



IntechOpen

Computer and Network Security

Edited by Jaydip Sen



Anomaly-Based Intrusion Detection System

Veeramreddy Jyothsna and Koneti Munivara Prasad

Abstract

Anomaly-based network intrusion detection plays a vital role in protecting networks against malicious activities. In recent years, data mining techniques have gained importance in addressing security issues in network. Intrusion detection systems (IDS) aim to identify intrusions with a low false alarm rate and a high detection rate. Although classification-based data mining techniques are popular, they are not effective to detect unknown attacks. Unsupervised learning methods have been given a closer look for network IDS, which are insignificant to detect dynamic intrusion activities. The recent contributions in literature focus on machine learning techniques to build anomaly-based intrusion detection systems, which extract the knowledge from training phase. Though existing intrusion detection techniques address the latest types of attacks like DoS, Probe, U2R, and R2L, reducing false alarm rate is a challenging issue. Most network IDS depend on the deployed environment. Hence, developing a system which is independent of the deployed environment with fast and appropriate feature selection method is a challenging issue. The exponential growth of zero-day attacks emphasizing the need of security mechanisms which can accurately detect previously unknown attacks is another challenging task. In this work, an attempt is made to develop generic meta-heuristic scale for both known and unknown attacks with a high detection rate and low false alarm rate by adopting efficient feature optimization techniques.

Keywords: intrusion detection, data mining, classification based, DoS, Probe, U2R, R2L, false alarm rate, zero-day attacks

1. Introduction

1.1 Internet security

Today, the world has numerous inventions and technological developments with proliferation of the Internet. Advances in business forced the organizations and governments worldwide to invent and use sophisticated and modern networks. These networks mix a variety of security aspects such as encryption, data integrity, authentication, and technologies like distributed storage systems, voice over Internet protocol (VoIP), wireless access, and web services.

Enterprises are more available to these systems. For instance, numerous business associations enable access to their administration on the system through intranet and web to their partners; endeavors empower clients to connect with the systems by means of web-based business exchanges that enable representatives to get to

data by methods for virtual private systems. This usage makes it more vulnerable to attacks and intrusions. A security threat comes not only from the external intruders but also from internal user in the form of abuse and misuse. A firewall simply blocks the network but cannot protect against intrusion attempts. In contrast, intrusion detection system (IDS) can monitor the abnormal activities on the network.

1.2 Intrusion detection systems (IDS)

Intrusion detection systems play a vital role in research and development with an increase in attacks on computers and networks [1]. Intrusion detection systems monitor the events occurring in a computer system or networks for analyzing the patterns of intrusions. IDS examine a host or network to spot the potential intrusions. Host-based systems explore the system calls and process identifiers mainly related to the operating system data. On the other hand, network-based systems analyze network-related events like traffic volume, IP address, service ports, and protocol used. Intrusion detection systems will

- i. analyze and monitor the system and user activities;
- ii. assess the integrity of critical system and data files; and
- iii. provide statistical analysis of activity patterns.

1.3 Taxonomy of intrusion detection systems

The intrusion detection systems are broadly classified as

- i. misuse detection systems and
- ii. anomaly-based detection systems.

1.3.1 Misuse detection systems

A misuse detection system is also called as signature-based detection that uses recognized patterns [2]. These patterns describe suspect, collection of sequences of activities or operations that can be possibly be harmful and stored in database. It uses well-defined patterns of the attack that exploits the weaknesses in system. The time taken to match with the patterns stored in the database is minimal. A key benefit of these systems is that the patterns or signatures can easily develop and understand the network behavior if familiar. It is more efficient to handle the attacks whose patterns are already maintained in the database.

The major restriction of these signature-based approaches is that they can only detect the intrusions whose attack patterns are already stored in the database. For every attack, its signature is to be created. Attacks whose patterns are not present in the database cannot be detected. Such technique can be easily deceived as they are dependent on a specific set of expressions and string matching. In addition, the signature works well only against fixed behavioral patterns; they fail to handle the attacks with human interference or attacks with inherent self-modifying behavioral characteristics.

These detection systems are also ineffective in cases where client works on new technology platforms such as no operation (NoP) generators, encoding, and decoding payloads. The efficiency of the signature-based systems decreases due to the need of creating dynamic signatures for different variations. With growing

volume of signatures, the performance of the engine also might lose the momentum. Because of this, intrusion detection frameworks are conducted on multiprocessors and Gigabit cards. IDS developers develop new signatures before the attackers develop solutions, in order to prevent any new kind of attacks on the system.

1.3.2 Anomaly-based detection systems

Network behavior is the major parameter on which the anomaly detection systems rely upon. If the network behavior is within the predefined behavior, then the network transaction is accepted or else it triggers the alert in the anomaly detection system [3]. Acceptable network performance can be either predetermined or learned through specifications or conditions defined by the network administrator.

The crucial stage of behavior determination is regarding the ability of detection system engine toward multiple protocols at each level. The IDS engine must be able to understand the process of protocols and its goal. Despite the fact that the protocol analysis is very expensive in terms of computation, the benefits like increasing rule set assist in lesser levels of false-positive alarms.

Defining the rule sets is one of the key drawbacks of anomaly-based detection. The efficiency of the system depends on the effective implementation and testing of rule sets on all the protocols. In addition, a variety of protocols that are used by different vendors impact the rule defining the process.

In addition to the aforesaid, custom protocols also add complexity to the process of rule defining. For accurate detection, the administration should clearly understand the acceptable network behavior. However, with strong incorporation of rules and protocol, the anomaly detection procedure would likely to perform more efficiently.

However, if the malicious behavior falls under the accepted behavior, in such conditions it might get unnoticed. The major benefit of the anomaly-based detection system is about the scope for detection of novel attacks. This type of intrusion detection approach could also be feasible, even if the lack of signature patterns matches and also works in the condition that is beyond regular patterns of traffic.

2. Network intrusion detection systems framework

In **Figure 1**, common intrusion detection framework (CIDF) integrated with Internet Engineering Tasks Force (IETF) and Intrusion Detection Working Group (IDWG) has successfully achieved efficient performance in representing the framework. This group defines a basic IDS structural design based on four functional modules.

Event modules (E-Modules) are defined as a combination of sensing elements and are engaged in continuous monitoring of the end system. In addition, these modules are also involved in processing the information events to the bottom three modules for further analysis.

Analysis modules (A-Modules) analyze the events and detect probable aggressive behavior, in order to ensure that some kind of alarm generated in essential conditions.

Data storage modules (D-modules) store the data from the E-Modules for further processing by the other modules.

Response modules (R-Modules) are used to provide the response to the transactions based on the information obtained from the analysis module.

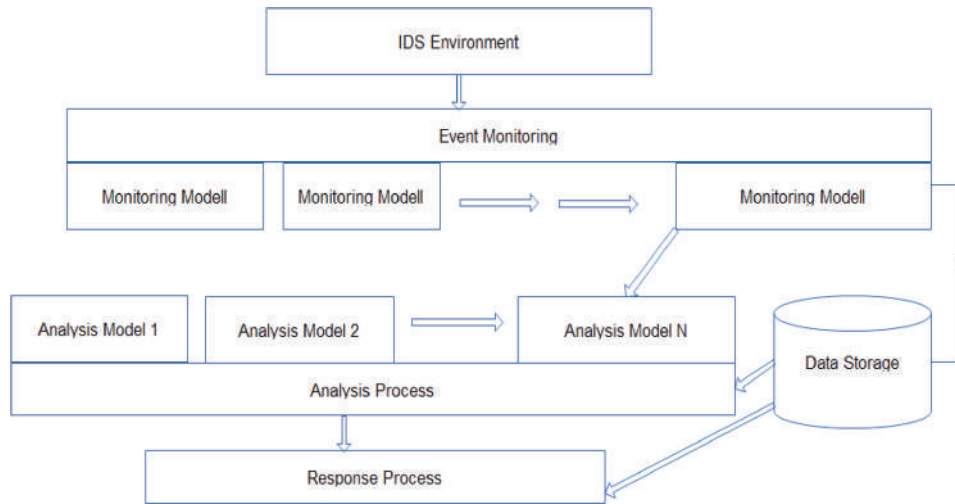


Figure 1.
Common intrusion detection framework architecture.

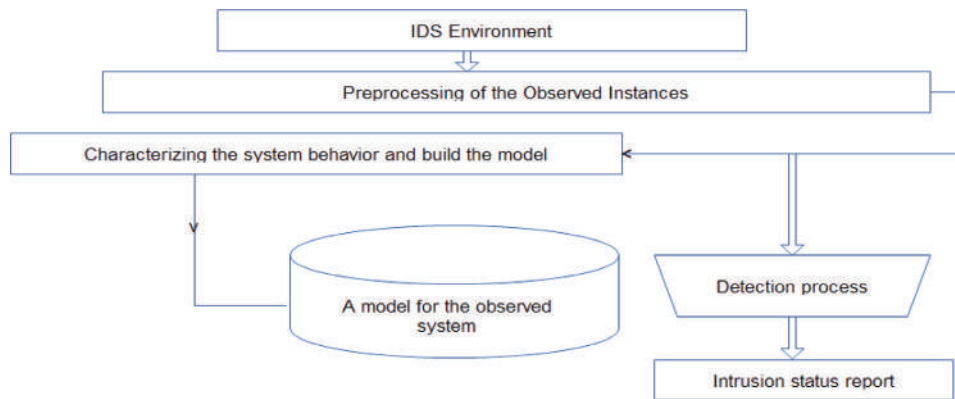


Figure 2.
Common anomaly-based network IDS.

Figure 2 represent the Common anomaly-based network IDS. The functional stages normally adopted in the anomaly-based network intrusion detection systems (ANIDS) are as follows:

Formation of attributes: In this stage, preprocessing of the attributes is done based on the target system.

Observation stage: A model that is built on the basis of behavioral features of the specified system where observations of intrusions can be carried out either through automatically or by manual detection procedure.

Functional stage: It is also called as detection stage. If the characterizing system model is available, it will match with the observed traffic.

3. Anomaly-based intrusion detection techniques

Figure 3 represents the taxonomy of anomaly-based intrusion detection techniques. They are statistical based, cognitive based or knowledge based, machine learning or soft

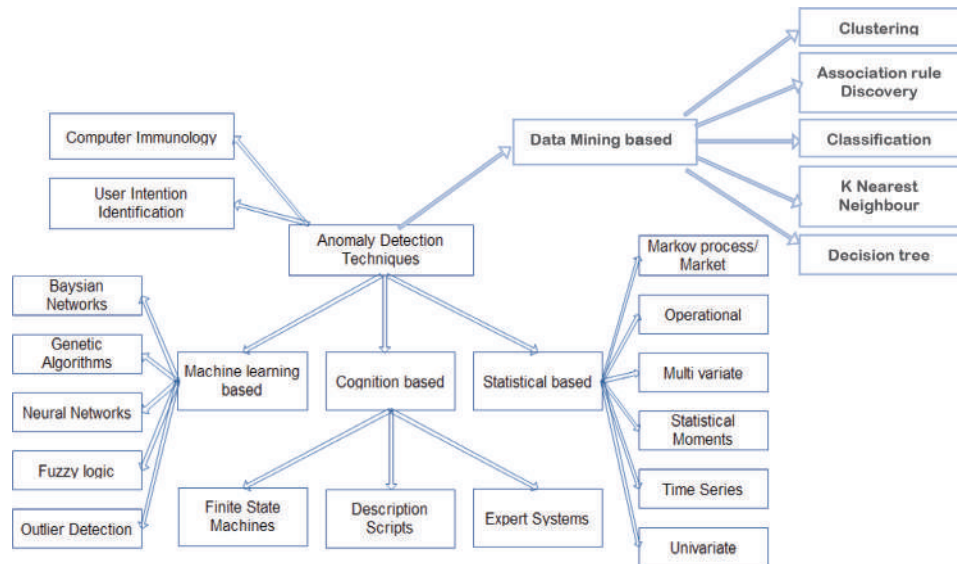


Figure 3.
 Classification of anomaly-based intrusion detection techniques.

computing based, data mining based, user intention identification, and computer immunology.

3.1 Statistical-based techniques

Statistical-based techniques use statistical properties such as mean and variance on normal transaction to build the normal profile [4]. The statistical tests are employed to determine whether the observed transaction deviates from the normal profile. The IDS assigns a score to the transactions whose profile deviates from the normal. If the score reaches the threshold, alarm is raised. The threshold value is set based on count of events that occur over a period of time.

Statistical-based techniques are further classified into operational model or threshold metric, time series model, Markov process model or Marker model, parametric approaches, statistical moments or mean and standard deviation model, multivariate model, and nonparametric approaches.

The main advantages of statistical-based techniques are as follows:

- i. They do not require any prior knowledge about the signatures of the attacks. So, they can detect zero-day attacks.
- ii. As the system is not depended on any of the signatures, updating is not required. Hence it is easy to maintain.
- iii. The intrusion activities that were occurred over extended period of time can be identified accurately and are good at detecting DoS attacks.

The disadvantages of statistical-based techniques are as follows:

- i. They need accurate statistical distributions.
- ii. The learning process of statistical-based techniques takes days or weeks to become accurate and effective.

3.2 Cognitive-based or knowledge-based techniques

Knowledge-based techniques are used to extract the knowledge from the specific attacks and system vulnerabilities. This knowledge can be further used to identify the intrusions or attacks happening in the network or system. They generate alarm as soon as an attack is detected. They can be used for both misuse and anomaly-based detection [5].

The knowledge-based techniques are broadly classified as state transition analysis, expert systems, and signature analysis.

The knowledge-based techniques possess good accuracy and very low false alarm rates. The knowledge gathered makes security analyst easier to take preventive or corrective action.

The knowledge-based techniques are maintaining the knowledge of each attack based on the careful and detailed analysis performed; it is a time-consuming task. A prior knowledge to update the each attack is a difficult task.

3.3 Data mining-based techniques

The knowledge-based IDS can detect the attacks whose patterns are known, but it is difficult to detect the inside attacks. One of the solutions is data mining techniques. The core idea is to extract the useful patterns and also the previously ignored patterns from the dataset [6].

The data mining-based techniques are further classified into clustering, association rule discovery, classification, K-nearest neighbor, and decision tree methods.

The key advantages of data mining-based techniques are as follows:

- i. They can handle high dimensional data.
- ii. As the precomputed models are designed in the training phase, comparing each instance at the testing phase can be done in faster way.
- iii. They can generate the patterns in unsupervised mode.

The key disadvantages of data mining-based techniques are as follows:

- i. These methods identify abnormalities as a by-product of clustering and as are not optimized for anomaly detection.
- ii. They require high storage and are slow in classifying due to high dimensionality.

3.4 Machine learning or soft computing-based techniques

Machine learning can be characterized as the capacity of a program or potentially a framework to learn and improve their performance on a specific task or group of tasks over a time [7]. Machine learning strategies emphasize on building a framework that enhances its execution based on previous results, that is, it can change their execution strategy based on recently acquired data.

Machine learning-based techniques are broadly classified as Bayesian approaches, support vector machines, neural networks, fuzzy logic, and genetic algorithms. Their key advantage is flexibility, adaptability, and capture of interdependencies. The disadvantage is high algorithmic complexity and long training times.

3.5 User intention identification

Intrusion detection system can be built based on the features that categorize the user or the system usage, to distinguish the abnormal activities from normal activities. During the early investigation of anomaly detection, the main emphasis was on profiling system or user behavior from monitored system log or accounting log data. The log data or system log may contain UNIX shell commands, system calls, key strokes, audit events, and network packages used.

3.6 Computer immunology

Computer immunology is a field of science that includes high-throughput genomic and bioinformatics approaches to immunology. The main objective is to convert immunological data into computational problems, solve these problems using statistical and computational approaches, and then convert the results into immunologically meaningful interpretations.

4. NSL-KDD dataset

The NSL-KDD [8] dataset is a refined version of its predecessor KDD99 dataset. NSL-KDD dataset comprises close to 4,900,000 unique connection vectors, where every connection vector consists of 41 features of which 34 are continuous features and 07 are discrete features. Each vector is labeled as either normal or attack. There are four major categories of attacks labeled in NSL-KDD: denial of service attack, probing attack, users-to-root attack, and remote-to-local attack.

- i. **Denial of service attack (DoS):** Denial of service is an attack category, which exhausts the victim's assets, thereby making it unable to handle legitimate requests. Examples of DoS attacks are "teardrop," "neptune," "ping of death (pod)," "mail bomb," "back," "smurf," and "land."
- ii. **Probing attack (PROBE):** Objective of surveillance and other probing attacks is to gain information about the remote victim. Examples of probing attacks are "nmap," "satan," "ipsweep," and "portsweep."
- iii. **Users-to-root attack (U2R):** The attacker enters into the local system by using the authorized credentials of the victim user and tries to exploit the vulnerabilities to gain the administrator privileges. Examples of U2R attacks are "load module," "buffer overflow," "rootkit," and "perl."
- iv. **Remote-to-local attack (R2L):** The attackers access the targeted system or network from the remote machine and try to gain the local access of the victim machine. Examples of R2L attacks are "phf," "warezmaster," "warezclient," "spy," "imap," "ftp write," "multihop," and "guess passwd."

5. Issues and challenges in anomaly-based intrusion detection systems

Although many methods and systems have been developed by the research community, there are still a number of open research issues and challenges. Some of the research issues and challenges of AIDS are as follows:

- i. A network anomaly-based IDS should reduce the false alarm rate. But, totally mitigating the false alarm is not possible. Developing an intrusion detection system independent of the environment is another challenge task for the network anomaly-based intrusion detection system development community [9–13].
- ii. Developing a general methodology or a set of parameters that can be used to evaluate the intrusion detection system is another challenging task [12, 13].
- iii. When new patterns are identified in ANIDS, updating the database without compromise of performance is another challenging task [9, 13].
- iv. Another task to be addressed is to reduce the computational complexities of data preprocessing in the training phase and also in the deployment phase [9, 10].
- v. Developing a suitable method for selecting the attributes for each category of attack is another important task [9–11].
- vi. Identifying a best classifier from a group of classifiers that is nonassociated and unbiased to build an effective ensemble approach for anomaly detection is another challenge [9–11].

6. Feature optimization using canonical correlation analysis

The preprocessed set of network transactions are partitioned based on its labeling (“normal” transactions as one set, “DoS” transactions as the other set and similar other range of sets). Unique values of each feature value set $f_i v(NTS)$ in the resultant normal transactions set (NTS) and its percentage of coverage are:

$$f_i v = \{f_i(v_1, c_1), f_i(v_2, c_2), f_i(v_3, c_3), f_i(v_4, c_4), \dots, f_i(v_j, c_j)\} \quad (1)$$

The procedure for feature optimization for each attack A_k is as follows:

- i. Consider the transactions set $ts(A_k)$ denoting attack type A_k (as an example considers DoS as an attack).
- ii. For every feature $f_i(A_k)$, consider all the values as a set $f_i v(A_k)$. An empty set $\overline{f_i v}$ of size $|f_i v(A_k)|$ is created and fills it based on its coverage as $|f_i v(A_k)| \cong |\overline{f_i v}|$, in which $|f_i v(A_k)|$ denotes the size of the feature values set $off_i(A_k)$.
- iii. The process is used to generate the feature values vector $\overline{f_i v}$ of the NTS, such that $\overline{f_i v}$ is compatible to the “ $f_i v(A_k)$ ” toward size and that also represents the coverage ratio of the values in $f_i v(NTS)$.
- iv. The process is applied for all feature values set in network transactions of attack A_k .
- v. Find the canonical correlation between $f_i v(A_k)$ and $\overline{f_i v}$. If the resultant canonical correlation is less than the threshold or zero, then the feature

$f_i(A_k)$ can be considered as optimal toward assessing the scale of intrusion scope.

It is imperative from the implementation of the above procedure that optimal features of a specific attack A_k can be identified. Further, the optimal features are ordered using the canonical correlation values. The values with lower than threshold are considered as optional set of features. Reducing the features leads to lesser computational complexities to the minimal level. The optimal features shall be used for further assessing the impact scale intrusion of type A_k .

7. Feature association impact scale (FAIS)

The approach for measuring the proposed feature association support (fas) metric considers the network transaction of the training dataset. The feature categorical values used in the network transactions are in the form of two independent sets. These values are used to develop a duplex graph between them.

7.1 Assumptions

Let $\{f_1, f_2, f_3, \dots, f_n \forall f_i = \{f_{i1}, f_{i2}, \dots, f_{im}\}\}$ be the set of categorical features values used for forming the set of network transactions T . Here T is a set of network transaction records of the given training set such as:

$$T = \{t_1, t_2, t_3, \dots, t_n \forall t_i = \{val(f_1), val(f_2), \dots, val(f_i), val(f_{i+1}), \dots, val(f_n)\}\} \quad (2)$$

Categorical values of the set of features related to every network transaction shall be considered as transaction value set tv_s and all transaction value sets are treated as “STVS.”

In the description above in Eq. 2, $val(f_i)$ can be expressed as $val(f_i) \in \{f_{i1}, f_{i2}, \dots, f_{im}\}$. The term “feature” refers to the current categorical value of the feature. The two features “ $val(f_i)$ ” and “ $val(f_j)$,” “ $val(f_i)$ ” are connected with “ $val(f_j)$ ” if and only if $(val(f_i), val(f_j)) \in tvs_k$.

7.2 Algorithm for FAIS technique

Step 1: The edge weight between the features $val(f_1)$ and $val(f_2)$ is estimated as:

$$w(val(f_1) \leftrightarrow val(f_2)) = \frac{ctvs}{|STVS|} \quad (3)$$

Step 2: The edge weight between transaction value sets and its corresponding set of feature categorical values can be measured as:

$$E = \{(tvs_i, val_j) : val_j \in tvs_i, tvs_i \in STVS, val_j \in v\} \quad (4)$$

Step 3: Further assuming the transaction value sets of the given duplex graph as pivots and the feature categorical values as pure prerogatives, the pivot and prerogative values are measured.

Step 3.1: Consider matrix u , which denotes pivot initial value as 1.

Step 3.2: Transpose the matrix A as A' .

Step 3.3: Calculate prerogative weights by multiplying A' with u .

Step 3.4: Calculate original pivot weights using matrix multiplication between A and V.

Step 4: Calculate the feature categorical value fas of $f_i v_j$ as:

$$fas(f_i v_j) = \frac{\sum_{k=1}^{|STVS|} \{u(tvs_k) : (f_i v_j \rightarrow tvs_k) \neq 0\}}{\sum_{k=1}^{|STVS|} u(tvs_k)} \quad (5)$$

Step 5: the Feature Association Impact Scale $fais$ for every transaction value set tvs_i is estimated as:

$$fais(tvs_i) = 1 - \frac{\sum_{j=1}^m \{fas(\{val_j \exists val_j \in V\}) : (val_j \subset tvs_i)\}}{|tvs_i|} \quad (6)$$

Step 6: The Feature Association Impact Scale threshold $faist$ can be measured as:

$$faist = \frac{\sum_{i=1}^{|STVS|} fais(tvs_i)}{|STVS|} \quad (7)$$

Step 7: Calculate the standard deviation as:

$$sdv_{faist} = \sqrt{\frac{\left(\sum_{i=1}^{|STVS|} fais(tvs_i) - faist^2\right)}{(|STVS| - 1)}} \quad (8)$$

Step 8: The Feature Association Impact Scale range can be explored as Step 8.1 and Step 8.2:

Step 8.1: Calculate lower threshold of $faist$ as $faist_l = faist - sdv_{faist}$.

Step 8.2: Calculate higher threshold of $faist$ as $faist_h = faist + sdv_{faist}$.

8. Analysis of experimental results

The total number of records chosen for the test is 25% of the actual dataset, that is, 34,361. The combination of test records chosen is from various categories such as Probe, DoS, U2R, R2L, and Normal. The difference between CC average and standard deviation of CC is called as lower bound of CC threshold. The sum of CC average and standard deviation of CC is called as upper bound of CC threshold.

The records that identified to be normal are 19.8% of the total test data records, with observations of 4.7% of it as “false negatives” and 15.1% of it as “true negatives.” The cumulative number of records that are detected as “intruded transactions” is 80.2%, with 75.3% of them being “truly intruded transactions” of test data records and the “false positive” percentage of 4.9% of test data records.

As per the results obtained, the proposed model is found to be accurate up to 90.4%. The experiments are conducted on the same dataset using “anomaly-based network intrusion detection through assessing Feature Association Impact Scale (FAIS)” [14]. The results depict that the proposed model is also scalable and

effective for detecting the scope of intrusion from a network transaction. Despite the fact that the FAIS model proposed shows 88% accuracy, the major limitation is process complexity in training the system. Such process complexities of designing the scale using FAIS are due to the number of features selected for assessing the scale. The issue of selecting the optimal features for training the Intrusion Detection System using Association Impact Scale is significantly addressed in the FCAAIS [15] model.

Table 1 indicates the comparison of performance metrics such as precision, recall/sensitivity, specificity, accuracy, and F-measure of FCAAIS over FAIS. **Figure 4** indicates that the accuracy of FCAAIS with optimal features is 91%, whereas the FAIS accuracy with all features is 88%. The precision of the FCAAIS model with optimal features and FAIS with all features is 92%. The other performance metrics such as sensitivity, specificity, and F-measure is calculated on FCAAIS over FAIS. The sensitivity, specificity, and F-measure are 96, 49, and 95%, respectively, for FCAAIS, whereas sensitivity, specificity, and F-measure are 95, 46, and 91%, respectively, for FAIS.

		FCAAIS	FAIS
	Total number of records tested	34,361	34,361
TP (true positive)	The number of transactions identified as normal, which are actually normal	29,379	27,889
FP (false positive)	The number of transactions identified as normal, which are actually intruded	1968	2752
TN (true negative)	The number of transactions identified as intruded, which are actually intruded	1901	2375
FN (false negative)	The number of transactions identified as intruded, which are actually normal	1113	1345
Precision	$TP/(TP + FP)$	0.937218873	0.910185699
Recall/sensitivity	$TP/(TP + FN)$	0.963498623	0.953991927
Specificity	$TN/(FP + TN)$	0.491341432	0.46323386
Accuracy	$(TP + TN)/(TP + TN + FP + FN)$	0.910334391	0.880765985
F-measure	$2 \times (PRECISION \times RECALL) / (PRECISION + RECALL)$	0.951646837	0.91131588

Table 1.
Comparison of performance metrics of FCAAIS and FAIS.

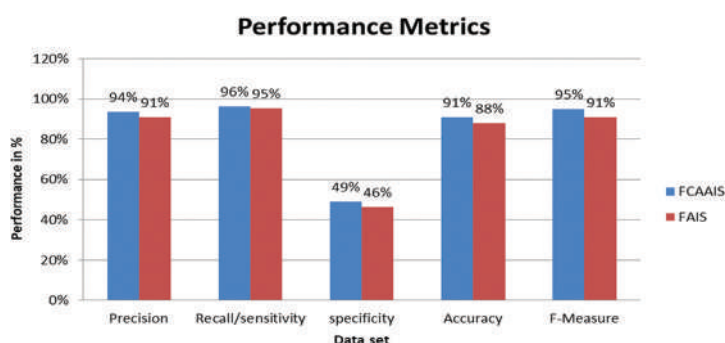


Figure 4.
The performance metrics observed for FCAAIS over FAIS.

According to the results, the accuracy of FCAAIS (selected feature set using canonical correlation) minimized the process complexity of designing the scale using FAIS (Figure 5 and Table 2).

The observed time complexity is adaptable, as the completion time is not directly related to the ratio of features count, which is due to the higher CC threshold as shown in Figure 6. Hence it is obvious to conclude that the applying canonical correlation toward optimized attribute selection is significant improvement to the FAIS model (shown in Figure 6).

It is observed that applying canonical correlation toward optimized attribute selection results in 3% improvement in the accuracy of FAIS [14]. Table 3 indicates precision, recall, and F-measure values calculated under divergent canonical correlation threshold values (Figure 7).

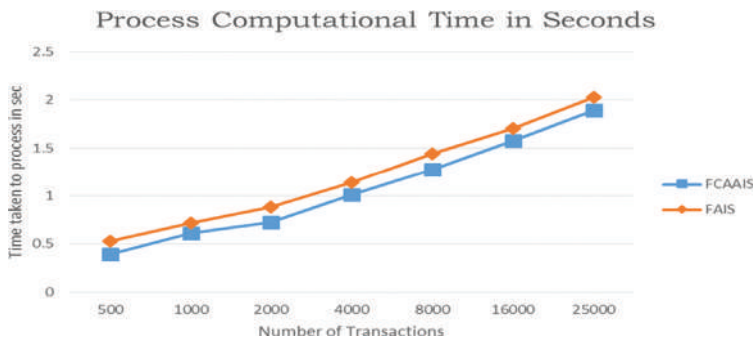


Figure 5.
The process computational time observed for FCAAIS over FAIS.

Number of transactions	FCAAIS (s)	FAIS (s)
500	0.397	0.527
1000	0.611	0.714
2000	0.723	0.882
4000	1.012	1.139
8000	1.275	1.439
16,000	1.578	1.703
25,000	1.891	2.031

Table 2.
Process computational time of FCAAIS and FAIS.

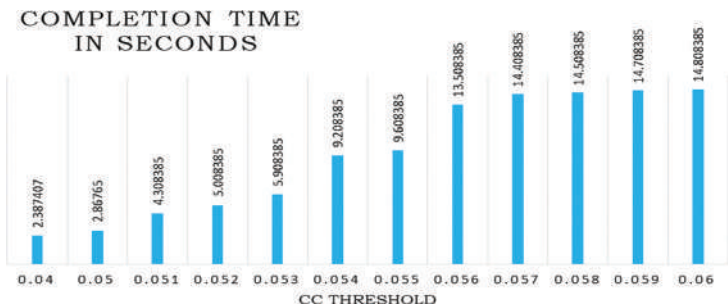


Figure 6.
The FCAAIS consumption of time under divergent canonical correlation thresholds.

	Precision	F-measure	Recall
Less than the upper bound of CC threshold	0.989	0.987998988	0.987
Less than the lower bound of CC threshold	0.98	0.984974619	0.99
Less than the CC threshold	0.985	0.985	0.985

Table 3.

Precision, recall, and F-measure values calculated under divergent canonical correlation threshold.

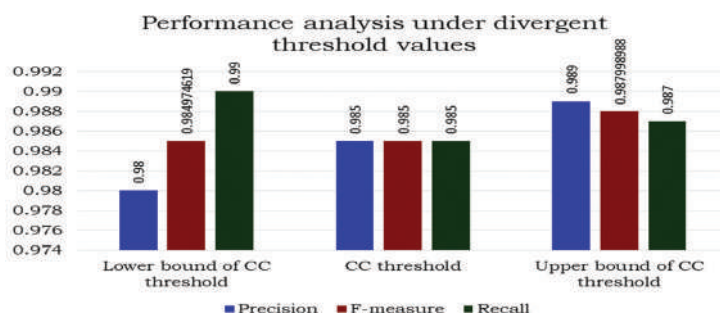


Figure 7.

Performance analysis of the prediction accuracy of FCAAIS under divergent canonical correlation threshold value.

9. Conclusion

It is desirable for anomaly-based network intrusion detection system to achieve high classification accuracy and reduce the process complexity of extracting the rules from training data. In this chapter, a canonical correlation analysis is proposed to optimize the features toward designing the scale to detect the intrusions. The selection of optimal features simplifies the process of FAIS. The experiments were conducted using a benchmark NSL-KDD dataset. The results indicate that the accuracy of FCAAIS with optimal features is 91%, whereas the FAIS accuracy with all features is 88%. The precision of the FCAAIS model with optimal features and FAIS with all features is almost close to 92%. It is observed that applying canonical correlation toward optimized attribute selection has 3% improvement in the accuracy of FAIS. The other performance metrics such as sensitivity, specificity, and F-measure is calculated on FCAAIS over FAIS. The sensitivity, specificity, and F-measure are 96, 49, and 95%, respectively, for FCAAIS, whereas they are 95, 46, and 91%, respectively, for FAIS.

Author details


Veeramreddy Jyothsna^{1*} and Koneti Munivara Prasad²

1 Sree Vidyanikethan Engineering College, Tirupati, India

2 Chadalawada Ramanamma Engineering College, Tirupati, India

*Address all correspondence to: jyothsna1684@gmail.com

IntechOpen

© 2019 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Debar H, Dacier M, Wespi A. A revised taxonomy of intrusion-detection systems. *Annales des Telecommunications*. 2000;55(7–8): 361-337
- [2] Gong Y, Mabu S, Chen C, Wang Y, Hirasawa K. Intrusion detection system combining misuse detection and anomaly detection using genetic network programming. In: ICCAS-SICE. 2009
- [3] Hall J, Barbeau M, Kranakis E. Anomaly-based intrusion detection using mobility profiles of public transportation users. *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*. 2005
- [4] Lee J, Moskovics S, Silacci L. A survey of intrusion detection analysis methods. CSE 221, University of California, San Diego, Spring 1999
- [5] Prayote A. Knowledge-based anomaly detection [PhD dissertation]. School of Computer Science and Engineering, The University of New South Wales; 2007
- [6] Caulkins LTCBD, Lee J, Wang M. A dynamic data mining technique for intrusion detection systems. In: *Proceedings of the 43rd Annual Southeast Regional Conference (ACM-SE 43)*. Vol. 2. 2005. pp. 148-153
- [7] Tsai C-F et al. Intrusion detection by machine learning: A review. *Expert Systems with Applications*. 2009; 36(10):11994-12000
- [8] Revathi S, Malathi DA. A detailed analysis on NSLKDD dataset using various machine learning techniques for intrusion detection. *International Journal Engineering Research and Technology (IJERT)*. Dec 2013;2(12)
- [9] Bhuyan MH, Bhattacharyya DK, Kalita JK. Network anomaly detection: Methods, systems and tools. *IEEE Communications Surveys & Tutorials*. 2014;16(1):303-336
- [10] Wagh SK, Pachghare VK, Kolhe SR. Survey on intrusion detection system using machine learning techniques. *International Journal of Computer Applications (0975–8887)*. 2013;78(16): 30-37
- [11] Gilmore C, Haydaman J. Anomaly detection and machine learning methods for network intrusion detection: An industrially focused literature review. In: *International Conference Security and Management*; CSREA Press.
- [12] Pedro G-T. Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security*. 2009;28(1):18-28
- [13] Patcha A, Park J-M. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer Networks*. 2007;51(12): 3448-3470
- [14] Veeramreddy J, Vaddella RPV. Anomaly-based network intrusion detection through assessing feature association impact scale. *International Journal of Information and Computer Security*. 2016;8(3):241-257
- [15] Jyothsna V, Rama Prasad VV. FCAAIS: Anomaly based network intrusion detection through feature correlation analysis and association impact scale. *ICT Express*. 2016;2(3): 103-116

Security in Wireless Local Area Networks (WLANs)

Rajeev Singh and Teek Parval Sharma

Abstract

Major research domains in the WLAN security include: access control & data frame protection, lightweight authentication and secure handoff. Access control standard like IEEE 802.11i provides flexibility in user authentication but on the other hand fell prey to Denial of Service (DoS) attacks. For Protecting the data communication between two communicating devices—three standard protocols i.e., WEP (Wired Equivalent Privacy), TKIP (Temporal Key Integrity Protocol) and AES-CCMP (Advanced Encryption Standard—Counter mode with CBC-MAC protocol) are used. Out of these, AES-CCMP protocol is secure enough and mostly used in enterprises. In WLAN environment lightweight authentication is an asset, provided it also satisfies other security properties like protecting the authentication stream or token along with securing the transmitted message. CAPWAP (Control and Provisioning of Wireless Access Points), HOKEY (Hand Over Keying) and IEEE 802.11r are major protocols for executing the secure handoff. In WLANs, handoff should not only be performed within time limits as required by the real time applications but should also be used to transfer safely the keying material for further communication. In this chapter, a comparative study of the security mechanisms under the above-mentioned research domains is provided.

Keywords: WLAN security, WEP, WPA, 802.11i, denial of service (DoS), lightweight authentication, secure handoff

1. Introduction

Wireless Local Area Networks (WLANs) provide an extension to the wired network. The wireless stations (STAs) connect to an Access Point (AP) for communication. The messages involved in the communication between STA and AP are visible to other STAs lying in the communication range. This makes WLANs insecure and hence WLANs requires protection.

As with any other computer network, the major security goals in WLANs are: confidentiality, integrity and availability (termed as CIA triad). Prominent techniques that help in attaining these goals include: access control, authentication, encryption, message authentication codes (MAC). Under Access control domain, the entity authentication is performed initially. Depending upon the entity authentication results, access into the WLAN network is controlled. For controlling access into the WLANs IEEE 802.11i (WPA2) is the main standard [1]. This standard though provides flexibility in user authentication but has several issues under the Denial of Service (DoS) attacks [2]. For providing protection to individual WLAN data frames encryption mechanisms like WEP (Wired Equivalent Privacy),

TKIP (Temporal Key Integrity Protocol) and AES-CCMP (Advanced Encryption Standard—Counter mode with CBC-MAC protocol) are used. Lepaja et al. [3] have demonstrated through experiment that WPA with AES provides high TCP throughput. Also, AES-CCMP protocol provides strong security properties, and hence is mostly used in the enterprises [3]. In WLANs, sometimes handoff by the STA is required to maintain communication continuity. There exist several protocols like CAPWAP (Control and Provisioning of Wireless Access Points), HOKEY (Hand Over Keying) and IEEE 802.11r that claim safe and continuous handoff by the STAs [4]. These protocols transfer safely the keying material to STA for further communication. The time limit constraint is imposed on such handoff as the handoff should be performed within short interval required by the real time applications.

This chapter is further divided into four sections. Section2 discusses access control methodologies in WLANs while section3 provides understanding of frame authentication methodologies. Section 4 explains secure handoff methods along with the requirements of secure handoff in WLAN environment. Each of these sections also provides comparative analysis among various methodologies. Section5 provides conclusions and future directions.

2. Access control

Traditionally, the entity authentication and access control is provided by the legacy authentication standard i.e., WEP. It has proved insufficient [2] and is hence, deprecated. Currently, IEEE 802.11i (WPA2) [1] security standard is used as an entity authentication and access control mechanism. This security standard is used to secure data communication over 802.11 wireless LANs. The IEEE 802.11i authentication specifies 802.1X authentication mechanism for large networks. The 4-way handshake follows an 802.1X authentication process to confirm the shared keys on Wireless Station (STA) and AP, evolving alongside the Pairwise Transient Key (PTK). This key is used to secure the data sessions between STA and AP using either Temporal Key Integrity Protocol (TKIP) or Advanced Encryption Standard (AES) in counter mode with a Cipher Block Chaining Message Authentication Code (CBC-MAC) Protocol (CCMP). As per the findings of Asante and Akomea-Agyin, use of simple passwords/passphrases makes CCMP susceptible to dictionary attacks [5]. The authentication and 4-way handshake are performed sequentially in 802.11i. Once STAs are authenticated, the standard evolves fresh secret keys to secure data communication over 802.11 wireless LANs. A large numbers of packets are used in these processes [2], which results in an increased process length, communication overhead and network overhead. The authentication and 4-way handshake both are prone to Denial of Service (DoS) attacks. This is due to the lack of proper authentication and insecure message communications between wireless devices [2, 6].

In 802.11i based Networks, 4-way handshake is used for evolving and sharing the keys between the two communicating partners. This 4-way handshake is one of the major concerns in WPA2/802.11i because of Denial of Service (DoS) attacks and therefore researchers target to reduce the 4-way handshake latency. Some suggested to make it 3-way while other suggested to make it 2-way [7]. One such improvement is proposed by Singh and Sharma [7]. In their proposal, the authors try to eliminate the entire 4-way handshake while maintaining the security and key refreshing requirements. For their purpose, they have utilized frame sequence numbers and the striking feature of the proposal is that the key freshness is maintained for each communicating frame. The key refreshed is used for fulfilling the security aspects like frame encryption and integrity management. The overheads in the proposal are bare minimum and it is lightweight as no changes in the existing MAC frame

are done. Also, no extra messages are required. Their improvement is more useful under frequent key refreshing situations where users are joining and leaving the wireless environment frequently like in a short duration conference/workshop or in lounge of railway station/airport. The improved technique provides a secure authentication mechanism and no explicit synchronization is required in case of loss of frames. The timings analysis done in the work shows that this technique is effective while security analysis shows that it enjoys almost equivalent security as compared with 4-way handshake of 802.11i. Removal of handshake ensures that the attacks conducted in the 4-way handshake are also removed.

Another improvement in the 802.11i standard is proposed by Singh and Sharma [8] wherein a novel sequence number based scheme is proposed to reduce the MIC field overhead in the WLANs. The existing security frameworks (WPA, 802.11i) provide MIC for maintaining the integrity and authentication for each data frame. MIC is kept in separate field in the frame, and hence adds to the communication overhead. The scheme of Singh and Sharma [8] introduces the notion of authentication token (AT). This AT is calculated based upon the existing sequence number of the WLAN frame. The AT serves both frame integrity and frame authentication purposes. After calculation, it is placed instead of sequence number in the sequence number field of the WLAN frame which means no extra bit or field overhead involvements. As MIC field is removed and AT placement requires no overheads, the scheme is effective as far as WLAN communication overheads and space managements are considered. In addition, the authors have shown that their method is resistant against replay attacks and also provided details on how to attain synchronization in case of frame loss.

In October 2017, a new and major weakness was documented in WPA2 WLAN standard termed as Key Reinstallation AttaCK or KRACK [9]. It was noted that this affected all kinds of WLAN security and hence the reputation of WPA2 got decreased. The WPA2 standard also suffered under DoS attacks. Hence, Wi-Fi Alliance comes up with the improvement. The improvement is termed as WPA3. Its main features involve: (1) ease of use (2) natural password selection (3) an improved and robust handshake and, (4) forward secrecy. The WPA3 is backward compatible with WPA2 which means the upgraded devices can work in WPA2 or WPA3 modes [10]. The market adoption of this standard is now picking and it will take some more time for getting stabilized. Thus, this work on WLAN security considers the present widespread standard i.e., WPA2.

Li et al. proposed an initial entity authentication scheme termed as fast WLAN initial access authentication protocol (FLAP) [11]. FLAP is targeted towards making access authentication faster by reducing the number of initial authentication messages. It is assumed in the protocol that STA and AS share common secret key which simplifies the entire mechanism. Overall, this method involves 6 messages (approx. Two round trip times, **Figure 1**), proves STA authentication at the AS via shared key, has key hierarchy equivalent to 802.11i and protects the messages by MIC. Through practical measurements it is shown that FLAP can improve the efficiency of EAP-TLS by 94.7 percent. It is suggested that this method is compatible with 802.11i and can coexist with existing 802.11i standard. Depending upon circumstances either 802.11i or FLAP can be chosen from suite selector. Like standard 802.11i security protocol, FLAP scheme also depends upon MIC for frame integrity and authentication despite of the fact that MIC verification is computation intensive. This protocol hence may fall an easy prey to Denial of Service (DoS) attacks wherein the attacker may send large number of frames having incorrect MICs. The successive MIC failures on the receiver results in a kind of DoS attack termed as computation DoS attack [12].

Singh and Sharma [13] proposed an access control authentication scheme—SWAS (Secure WLAN Authentication Scheme). The scheme introduces the concept

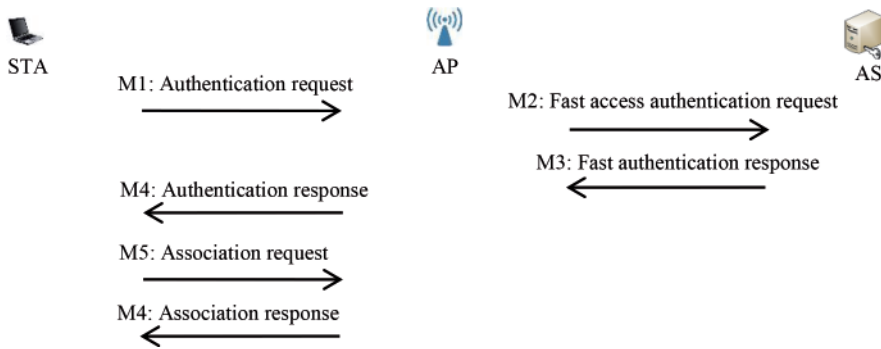


Figure 1.
A simplified overview of initial access authentication protocol (FLAP).

of delegation in WLANs and provides access to clients only upon authentication. SWAS provides authentication of all parties (STA, AP and AS) and evolves a fresh key for securing the data sessions. In addition, it provides security to all messages by utilizing cryptographic primitives, such as encryption and Message Integrity Code (MIC). The proposed scheme reduces the length and complexity compared to IEEE 802.11i authentication and key deriving process. The use of cryptographic techniques does not increase the authentication time of the proposed method. The scheme reduces the communication cost, network overhead and is also resilient against DoS attacks. Therefore, the main contribution of SWAS is to provide a secure and efficient authentication mechanism that evolves fresh communication keys.

The SWAS scheme involves three parties: STA, AP and AS. It has three phases: registration phase, request phase and authentication phase. Initially, STA registration is performed at AS and is required only once in a given network. In registration, AS utilizes delegation concept, and generates shared secret key (σ) for AS and STA [14]. The registration phase is followed by the request phase, where the existing 802.11 probe requests, and the probe response messages are utilized by the STA to request the network connection and access. After the request phase, SWAS authentication is performed for authentication and to derive a new communication key that is used to protect the data packets in subsequent sessions.

Both online and offline authentications are used in the SWAS scheme. Online authentication provides authentication and security to all messages among STA, AP and AS. The online authentication utilizes three random numbers (r_1 , r_2 , r_3) and a sequence number (s_1) to ensure proper encryption, authentication and key freshness. In addition, it maintains a key hierarchy similar in purpose to 802.11i with a Master Session Key (MSK), Pairwise Master Key (PMK) and Pairwise Transient Key (PTK). The PTK evolved on the STA and AP during the authentication process is used to encrypt the data packets between them. A simplified view of the SWAS online authentication message exchanges (M1, M2, M3 and M4) is shown in **Figure 2**. In this figure it is clearly visible that each one among STA, AP and AS authenticates each other through various passcode/digital signature verification. The passcode is nothing but protected information (secured through cryptographic means) for the other party. Offline authentication is required whenever a new session key between the same STA and AP is required. This does not involve AS for authentication rather it uses prior stored information at STA and AP. The offline authentication is done via a re-association request and utilizes loosely synchronized sequence number scheme [15].

The salient features of SWAS include: (1) Resistance to DoS attacks in almost all the phases, (2) Less communication and computation time as compared with

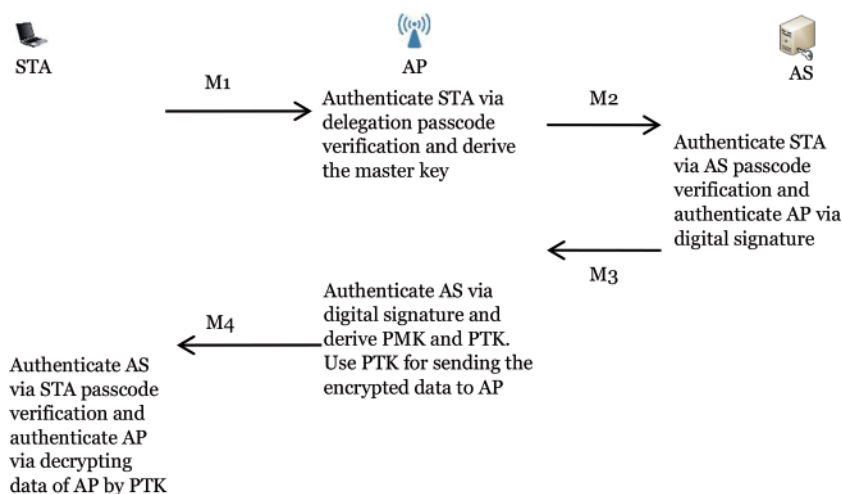


Figure 2.
 A simplified overview of online authentication phase of SWAS scheme.

IEEE 802.11i standard, (3) authentication of all the associated parties i.e., STA, AP and AS by each other and, (4) authentication of all the messages used during all the protocol communication phases. The shortcomings include: (1) lack of practical demonstration of the protocol and (2) no extension of the scheme under the handoff situations is provided till date.

Authentication per frame and symmetric key based encryption is an implicit necessity for security in Wireless Local Area Networks (WLANs). Singh and Sharma [16] proposed a novel symmetric key based Access Control and per frame authentication scheme for WLANs termed as Key Hiding Communication (KHC) scheme. KHC scheme has two phases: initial phase and communication phase. Former is utilized for sharing and evolving the master key (MK) between STA and AP whereas latter is utilized for onwards data frame communication using the (refreshed) keys. The major establishment of this scheme is the introduction of novel concepts of refreshing the key, protecting the key and initial vector (IV) using different counters and then mixing the bytes of protected key and IV together for each communicating frame. The mixing is based upon the shared secret key and hence only the two communicating parties i.e., STA and AP can mix and separate the bytes of key and IV. The protected mixed bytes are termed as codeword while the concept of mixing the protected key and IV bytes is termed as key hiding. The codeword is added in the WLAN frame. This addition of codeword to the existing WLAN frame occupies extra space and hence the scheme has extra space overheads. Integrity to the frame is provided via MIC. A new key and new IV for the new frame to be transmitted is evaluated based upon existing secret key and existing IV. Evaluation of new key and new IV is termed as key and IV refreshing. The refreshed new key and new IV are first protected using incremented values of counters and then mixed together to form new codeword. The verification and separation of the key and IV from the transmitted codeword provides frame authentication. Once the frame is authenticated, its integrity is verified through MIC verification involving key. The frame authentication is lightweight in KHC as it involves trivial increment, XOR and modulus operations. Thus, KHC follows the notion of frame authentication first and then checking the frame integrity for protection against computation DoS attacks. The separated key and IV are used to decrypt the frame contents and are also used to confirm the frame integrity via MIC. The simplified overview of KHC communication process is shown stepwise in **Figure 3**.

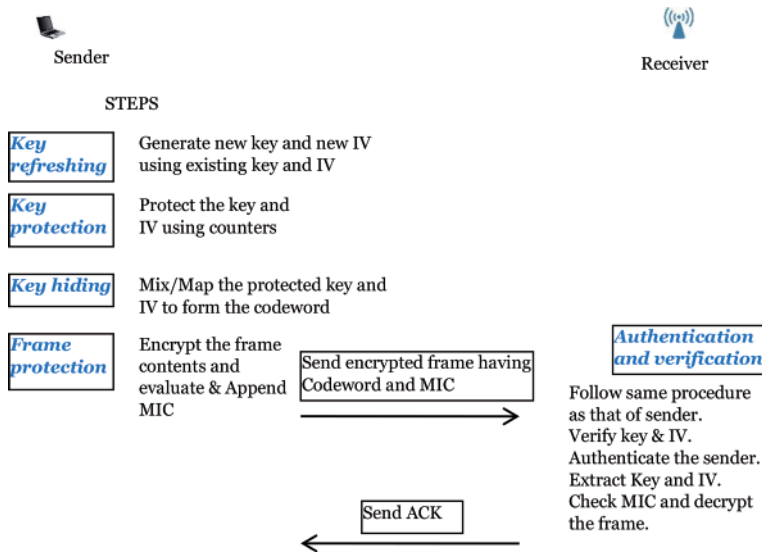


Figure 3.
A simplified overview of communication phase of KHC scheme.

In nutshell, KHC introduces the concept of key hiding which involves protecting the key using counters followed by mixing of refreshed key & IV i.e., mapping of refreshed key & IV. Through this process of formation of the codeword, the secret symmetric key remains concealed from the attacker. The recipient extracts the key from the codeword, compares it with its own evaluated key, thereby authenticating the sender. Key along with IV, is then used to decrypt the data frame of the sender. Thus, KHC is a useful WLAN communication scheme that is not only secure but is also efficient. The major contributions made by KHC are: (1) lightweight WLAN communication methodology, (2) utilization of symmetric key based encryption/decryption, (3) Per frame Key refreshment, (4) protection against computation DoS attacks and, (5) comparable security as that of 802.11i.

2.1. Comparisons of various WLAN access control mechanisms

A property wise comparison between prominent WLAN access control security mechanism is presented in **Table 1**. WEP is though deprecated but mentioned here for the sake of completeness. It can be noted that WEP provides weak authentication, integrity and encryption support. Further, WEP does not consider key and IV refreshing. IEEE 802.11i is a strong protocol as it maintains strong authentication, integrity and encryption. It involves large number of messages and hence consumes times during initial authentication. For key refreshing, it involves 4-way handshake having 4 message exchanges between STA and AP. This 4-way handshake is the major concern in 802.11i. It is prone to DoS attacks and KRACK attacks. FLAP and SWAS both enjoys features similar to that of 802.11i with a difference that the messages exchanged for symmetric key evaluation are less in FLAP and SWAS. In FLAP, very few i.e., approx. 6 messages are exchanged for the key evaluation (including those between STA and AP). In SWAS, only four (4) initial messages are required during online authentication (including those between STA and AP) for sharing the PTK. During offline authentication for refreshing the shared symmetric key only two messages are required. The KHC scheme adopts an interesting methodology which is different from the other access control protocol.

WLAN access control—Security mechanisms					
Property	WEP	802.11i [1]	FLAP [11]	SWAS [13]	KHC [16]
Authentication	Yes, weak	Yes, strong, initial entity authentication followed by MIC based per frame auth.	Yes, strong, initial entity authentication followed by MIC based per frame auth.	Yes, strong, initial authentication followed by MIC based per frame auth.	Yes, strong, initial entity authentication followed by continuous, lightweight per frame auth.
Integrity support	Yes, weak, CRC based	Yes, strong, MIC based	Yes, strong, MIC based	Yes, strong, MIC based	Yes, strong, MIC based
Encryption support for confidentiality, strength of encryption	Yes, low, RC4 algorithm	Yes, high, TKIP and AES based	Yes, high, TKIP and AES based	Yes, high (Once shared key is evolved, rest process is same as that of 802.11i)	Yes, high, any one of RC4/ TKIP/ AES can be used
Synchronization Algorithm	No	No	No	No	Yes
Initial message Exchange for symmetric key exchange	No, done manually	Yes, large	Yes, few – 06 messages (two round trip times)	Yes, few -only four (4) initial messages during online authentication	Yes, few
Key freshness	No	Yes	Yes	Yes	Yes
IV freshness	No	N.A.*	N.A.*	N.A.*	Yes
Messages exchange for key renewal	N.A.*	Yes, four, explicitly	Yes, four (between STA and AP), explicitly	Yes, two using offline authentication	No, done implicitly

*Not Applicable in this mechanism.

Table 1.
Property wise comparison of WLAN access control security mechanisms [16].

It does not use any third party like AS in the authentication process and hence involves less number of messages. It provides an implicit key hiding per frame authentication procedure that is capable of communicating the key to the other entity and is able to refresh not only the shared key but also the IV for encrypting each frame. Thus, least messages are required for key refreshing among all the access control WLAN security mechanisms. Also, the adopted methodology of key refreshing, protection and mapping makes the cracking of key difficult for the attacker. In contrast to WEP, IV is hidden and not visible to the attacker. Other access protocols do not have the notion of IV.

As shown in **Table 2**, memory requirements of WEP is least. 802.11i has more memory requirements than WEP but less than others. Among others, SWAS has highest while FLAP has lowest memory requirements. Communication overhead analysis shows that (1) KHC and WEP involves per frame overheads whereas in others it is done implicitly and, (2) KHC is efficient in key refreshing as compared to others. For key refreshing each of 802.11i and FLAP requires 4 frames,

WLAN access control—Security mechanisms					
Overheads	WEP	802.11i [1]	FLAP [11]	SWAS[13]	KHC[16]
Memory requirements **	Storing key and IV	Storing Master Key, Refreshed key	Storing Master Key, Refreshed key and counter	Storing delegation key, public key pairs, Symmetric keys: MK, PMK, MSK, PTK, two counters, one sequence number. (Also pool of random numbers at AP)	Storing Master Key, Refreshed key, IV and two counters
Communication overheads					
For per frame authentication	IV (128 bits) per frame	Implicitly by MIC	Implicitly by MIC	Implicitly by MIC/ authentication information	256 bits per frame
For key refreshing	N.A.*	4 data frames	4 data frames	2 data frames	implicit
*Not Applicable in the scheme. **Considered per participating node.					

Table 2.
Performance comparison of WLAN access control security mechanisms [16].

SWAS requires 2 frames whereas it is handled implicitly in KHC. In [11], the average authentication delays of the EAP-TLS and FLAP are evaluated as 260.253 and 13.884 ms, respectively. In [13], the total time for SWAS authentication is found to be of the order of 26.46 ms (including time for DoS protection). In [16] Key refreshing timings of 802.11i and KHC are shown as 13.5 ms and 7.5 ms, respectively.

The security comparison shown in **Table 3** clearly indicates that SWAS and KHC scheme provides almost equivalent and better security. 802.11i is prone to DoS attacks whereas FLAP is prone to replay and man-in-middle attacks. Obviously, security of FLAP is least and hence it is not much used presently.

In most of the WLAN access control mechanisms (except KHC), authenticity to the data frame is usually provided by MIC. The MIC based per frame authentication may lead to computation DoS. Hence, lightweight per frame authentication solution is required. It is discussed next.

Attacks	WEP	802.11i [1]	FLAP [11]	SWAS[13]	KHC [16]
Possibility of frame contents overwritten by attacker	Yes	No	No	No	No
Possibility of modification of authentication bits	N.A. as authentication is implicit	No	No	N.A.*	No
Man-in-middle attack	Yes	No	Yes	No	No
Replay attack	Yes	No	Yes	No	No
Reduce DoS attacks	No	No	No	Yes	Yes
*Not applicable in this mechanism.					

Table 3.
Comparison of WLAN access control security mechanisms under attacks [16].

3. Frame authentication

In WLANs, a two layer redundant security exists. One at the Medium Access Control (MAC) layer while other at the higher layer dealing with End to End security. In former, 802.11i provides security while in latter, higher layer protocols like IPSec, SSL-TLS etc. provides security. Hence, it is suggestive that lightweight authentication and symmetric key based cryptographic measures per frame should be used.

For providing individual frame level protection, two kinds of per frame authentication exist in WLANs: MIC based authentication and lightweight authentication. MIC based frame authentication for data frames is utilized by standard WLAN protocols like IEEE 802.11i, FLAP etc. In these protocols, each frame is accompanied by a unique MIC calculated using sender's shared secret key. The receiver verifies it by recalculating and matching using its share secret key. The MIC calculations and verification consume computation time of the order of 1.5 ms and as shown in Section 2 for FLAP protocol, computation DoS attacks are a possibility [12, 17, 18]. Main reason for computation DoS attack is attributed to the fact that MIC is serving two purposes: authentication and message integrity. Instead, first lightweight authentication should be used. If it succeeds, frame integrity (MIC) should be checked only for those frames whose authentication has succeeded. This will reduce the DoS attacker chances. Thus, lightweight authentication techniques which uses less computation time may prove useful.

The lightweight authentication schemes [19–25] generate the random authentication bits at sender and receiver using random bit generator with commonly shared secret seed as input. These authentication bits are inserted into the WLAN frames. Upon verification of the authentication bits, the frame is accepted at the receiver. Though such schemes provides authentication but they usually lack other security measures like key freshness, secrecy and integrity. A brief tabulation of these schemes is presentation in **Table 4**, showing advantage and disadvantage of each.

3.1 Comparisons of various lightweight authentication mechanisms

All the schemes considered in **Table 4** provide per frame continuous authentication. Schemes of Pepyne et al. [25] and Singh and Sharma [26] supports integrity. Former supports CRC based weak integrity while latter supports MIC based strong integrity. Schemes of Pepyne et al. [25] and Singh and Sharma [26] supports encryption. Former supports RC4 based weak encryption while latter supports TKIP/AES based strong encryption. All the schemes considered use their own synchronization algorithm, in fact scheme by Wang et al. [22] uses three different synchronization algorithms. Schemes by Ren et al. [23], Lee et al. [24], Pepyne et al. [25] and Singh and Sharma [26] involves initial message exchanges. Key freshness is incorporated by Pepyne et al. [25] and Singh and Sharma [26]. None of these involves extra messages for evolving new symmetric key (key renewal).

Considering the memory requirements of these schemes Singh and Sharma [26] has the greatest (912 bits) while Lee et al. [24] has the lowest (24 bits). Others except Pepyne et al. [25] have 256 bits memory requirements. Pepyne et al. [25] has 384 bits memory requirements. As far as communication overheads are concern, Johnson et al. [19, 20] and Ren et al. [23] have requirements of 3 bits per frame and 7 bits per ACK frame for counter. Wang et al. [21, 22] has no extra bit requirements as these keep the authentication bits in the unused type and subtype fields of 802.11 frame. Lee et al. [24] requires four extra frames, each having 3 authentication bits. Pepyne et al. [25] has requirements of keeping 128 bits per frame for keeping counter. ASN based scheme by Singh and Sharma [26] has no explicit requirements but requires 48 bits per ACK for synchronization.

Light weight authentication schemes	Features	Advantage(s)	Disadvantage(s)
Johnson et al. [19] Wu et al. [20]	Only one bit from the authentication stream generator is placed in the link layer data frame	<ul style="list-style-type: none"> • scheme provides originator sender identity authentication • has low communication overhead • as one bit can easily be damaged, synchronization algorithm is also proposed 	<ul style="list-style-type: none"> • attack leading to non-synchronization can easily be launched via successive frame authentication failures • The number of bits used for authentication purpose is too less due to which attacker has 50% chances • the data packets are not encrypted in SOLA nor MIC per frame is provided, hence payload may be changed (overwrite attack)
Wang et al. [21]	<ul style="list-style-type: none"> • the sender and the receiver generates an authentication stream using same seed value • The bit from the authentication stream is put in the frames by the sender and are verified by the receiver using its authentication stream 	lightweight protocol with synchronization algorithm and low communication overhead	<ul style="list-style-type: none"> • The authentication bits are not bound to the frame contents • synchronization process is affected by flooding DoS attack where the attacker confuses the sender via unauthenticated ACK frames • long authentication bits of continuous 0's or 1's by attackers in the frames can cause confusion
Wang, et al. [22]	<ul style="list-style-type: none"> • single bit lightweight authentication solution • Concept of discrimination among legitimate STAs and attacker nodes is used 	efficient in terms of computation cost, communication cost and synchronization efficiency	Possibility of authentication bit manipulation by attacker exists
Ren et al. [23]	3 bit authentication solution	Has synchronization algorithm that uses 7 bit counter value put in the ACK frame by the receiver for attaining synchronization	still utilizes less number of bits and therefore high probability of attacks
Lee et al. [24]	Scheme selects 3 bits for authentication of management frames	Protection from DoS attack performed by unauthenticated management frames	<ul style="list-style-type: none"> • scheme protects only the management frame whereas the data frame are not protected • DoS attack is still possible by using frames other than the management frames
Pepyne et al. [25]	<ul style="list-style-type: none"> • based upon improvising the WEP protocol • uses random stream generator for generating the authenticator variables and fresh encryption keys 	Frame counter 'k' is used for synchronization purpose	attacker can easily modify 'k' and launch the attack leading to non-synchronization and Denial of Service

Light weight authentication schemes	Features	Advantage(s)	Disadvantage(s)
Singh and Sharma [26]	<ul style="list-style-type: none"> utilizes sequence number of the frame along with the authentication stream generators for authentication provides authentication by modifying sequence number of the frame by trivial math operations by sender such that the modification is verified at the receiver 	<ul style="list-style-type: none"> it requires no extra bits or messages for authentication purpose and also no change in the existing frame format is required lightweight authentication helps in protecting against computation DoS attacks prohibits replays and maintains the synchronization 	AP maintains sequence numbers per STA

Table 4.
 Comparison of per frame WLAN authentication solutions.

On comparing the computational performance of the lightweight authentication schemes mentioned in **Table 4**, it is found that Pepyne et al. [25] and Singh and Sharma [26] take more computational time as compared with others. Singh and Sharma [26] takes more computational time due to the fact that it involves MIC evaluation and encryption of frame for enhancing the security. It is shown in [26] that considering only the authentication the time taken for computational cost for is 0.5 micro seconds which implies that it is same as that of other lightweight solutions.

Except, Pepyne et al. [25], the chances of Brute Force attacks on authentication bits embedded in the frames are quite high in these schemes. Except Pepyne et al. [25] and Singh and Sharma [26] the possibilities of frame contents modification, man-in-the middle attack, replay attacks and DoS attacks are quite high. Pepyne et al. [25] and Singh and Sharma [26] do not allow frame contents modifications and DoS attacks. Pepyne et al. [25] suffers under man-in-the middle attack and replay attacks.

Though KHC is considered in this chapter initially under the Access control mechanisms, it involves lightweight per frame authentication also and needs a special mention in this sub-section. In comparison with the schemes mentioned in **Table 4**, KHC has longer initial entity authentication process. KHC also has raised memory requirements but meets important security features like forward secrecy, key refreshing, lightweight per frame authentication, per frame encryption etc. required by any WLAN security protocol.

Apart from the two main authentication types i.e., MIC based authentication and lightweight authentication, the others are password key exchange mechanisms and layered authentication. The password key exchange mechanisms [27, 28] provide mutual authentication between client and authentication server (AS), identity privacy, half forward secrecy and low computation cost for a client. These mechanisms lack some of the mandatory and recommended requirements for the key exchange methods [29]. Also, these schemes provide authentication at the AS level only while ignoring the authentication at the AP level. The layered authentication achieved by EAP which acts as basis for higher layer authentication protocols, contains certain vulnerabilities e.g. no identity protection, no protected cipher suite negotiation, and no fast reconnection capability [29].

4. Secure handoff

WLANs handoffs are essential for providing continuous mobility to a wireless Station in an Enterprise LAN. Two important requirements of the handoff are: (1) establishment of a secure connection of the roaming STA with new access point (AP) and (2) completion of handoff within time limits such that the undergoing communication remains unaffected. The time limit on handoff for multimedia and real time WLAN applications is approximately 50 ms [30]. During this period no data packets transfer occurs. As per the 802.11i WLAN security standard, the complete secure STA authentication (default Full EAP/TLS) via AS evolving shared secret key between STA and AP takes time of the order of 300 ms to 4 s [12] and hence is unfit for the handoffs. For reducing this time, notion of pre-authentication is introduced wherein full 802.1X authentication involving AS is done utilizing old AP and candidate AP (new AP). Hence, at the time of handoff only 4-way handshake is required between STA and candidate AP. In this pre-authentication process, an inaccurate candidate AP prediction has associated resource wastage issues as full 802.1X will again be required [31]. Researchers have considered predictive authentication and proactive key distribution for reducing the handoff times. Former involves predicting the candidate AP whereas latter involves locating a group of candidate APs. Thus, in former the problem of inaccurate candidate AP prediction exists whereas in latter the problem of extra communication overhead for authentication with group of APs exists.

Researchers have also worked towards reactive solutions wherein the candidate AP is selected by STA and then the security context is transferred to this AP. In such solutions, STA requests to AS via old AP, then AS transfer security context and material to the candidate AP. Singh and Sharma [32] proposed one such novel secure handoff scheme that maintains security properties while evolving and transferring the security context (key and initial vector) to the candidate AP. The scheme is lightweight and uses reactive method for handoff. Two kinds of APs are defined in the scheme: normal AP and Domain Controller AP (DCAP). STA request DCAP through AP by putting ID of the candidate AP. DCAP in turn distributes the STA context (key and initial vector) to the candidate AP. Thus, when STA roams into the area of candidate AP, less time is involved in the STA authentication at the candidate AP.

For providing fast and secure handoff for the mobile STA in WLANs, standard bodies IEEE and IETF have defined protocols like Control and Provisioning of Wireless Access Points (CAPWAP), HandOver Keying (HOKEY) and IEEE 802.11r (Task group r) [5]. CAPWAP supports centralized management of APs. HOKEY extends the Authentication, Authorization and Accounting (AAA) architecture to support key deriving and distribution with involving full EAP authentication. 802.11r depends upon passing credentials directly between APs for handover. Though CAPWAP takes very less time, it is more or less re-authentication with centralized Access Controller (AC), followed by key transfer to new Wireless Termination Points (WTP). HOKEY is successful in multidomains but it takes more communication time. Among these three (CAPWAP, HOKEY and 802.11r), 802.11r is more efficient in terms of communication overheads. It still has issues concerning the safe transfer of key between APs.

4.1 Comparisons of various handoff mechanisms

CAPWAP and HOKEY does not change the existing 802.11 frame structure. 802.11r is a separate protocol and hence has different frame structure. All except CAPWAP scheme generates fresh session keys. Fresh traffic keys are generated by all the schemes. Communication overhead of KHC based handoff scheme is less as

compared to any other scheme. This handoff scheme shortens the handoff latency by initiating a key transfer process prior to moving to the new AP and performing handoff. It strengthens the security by (1) protecting STAs from re-associating to Malicious APs, (2) evolving fresh keys even during handshake, (3) authenticating all the frames during the handoff and, (4) safeguarding against DoS attacks and, (5) providing continuous authentication during communication.

5. Conclusions

This chapter discusses about the present WLAN security environment. It is clear that the WLAN security environment till date is dominated by WPA2 (IEEE 802.11i) standard. Researchers have pointed out regarding length and complexity of the WPA2. The major point of concern in WPA2 is key refreshing mechanism i.e., 4-way handshake due to which the WLAN security is considered vulnerable. Researchers, hence target to reduce the length of this handshake while maintaining the security properties intact.

The chapter also studies other WLAN security mechanisms proposed by researchers and categories them into: (i) access control, (ii) per frame authentication and (iii) secure handoff mechanisms. It provides category wise comparative analysis of these mechanisms. Three mechanisms are considered in the access control category. Among them Key Hiding Communication (KHC) is the most attractive but it requires changes in the existing WLAN frame structure. Per frame category is further sub-categorized into: (a) per frame authentication mechanisms utilizing MIC and (b) lightweight per frame authentication mechanisms. For enhancing the security, most of the per frame authentication solutions rely on MIC for both authentication and integrity of frame. It is shown that this MIC verification involves computation time and large number of such verifications may result in computation DoS attack on the receiver. The researchers hence advocate separating the authentication and integrity parts in per frame authentication. The lightweight per frame authentication mechanism are though lightweight in nature but lacks security properties like key refreshing, secrecy and integrity. In this chapter, several handoff mechanisms for WLAN environment are also discussed and it is accomplished that none guarantees to maintain required level of security during the specified handoff time limits.

WLAN security is having a transformation from WPA2 to WPA3. WLAN security is strengthened in the upcoming standard i.e., WPA3. It is very early to comment on the effectiveness of WPA3 and it is evident that the existing WLAN devices will continue to use WPA2. The new upcoming WLAN devices will obviously follow the backward compatibility towards WPA2. Thus, researchers can still target to test the implementation of 802.11i with the novel ideas like MIC reduction, 4-way handshake reduction and blockchain application in WLANs [33]. In wireless medium, per frame lightweight authentication mechanisms will prove an edge and in future, researchers may consider developing such solutions. For maintaining uninterrupted communication quick, secure, accurate and secure handoff is the need of the hour. Hence, researchers in future may consider implementation of efficient and secure handoff mechanisms using WPA3.

Acknowledgements

The authors acknowledge and express the gratitude towards their parent Institutes for the support.

Conflict of interest

The authors declare no conflict of interest.

Author details

Rajeev Singh^{1*} and Teek Parval Sharma²

1 G.B. Pant University of Agriculture and Technology, Pantnagar, Uttarakhand, India

2 National Institute of Technology, Hamirpur, Himachal Pradesh, India

*Address all correspondence to: rajeevpec@gmail.com

IntechOpen

© 2019 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



References

- [1] IEEE 802.11i. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Security Enhancements. 2004; IEEE Standard
- [2] Singh R, Sharma TP. On the IEEE 802.11i security: A denial of service perspective. *Wiley Journal of Security and Communication Networks*. 2015;8(7):1378-1407
- [3] Lepaja S, Maraj A, Efendiu I and Berzati S. The impact of the security mechanisms in the throughput of the WLAN networks. In: *Proceedings of the 7th Mediterranean Conference on Embedded Computing*; June 2018; Budva, Montenegro
- [4] Clancy TC. Secure handover in Enterprise WLANs: CAPWAP, HOKEY, and IEEE 802.11r. *IEEE Wireless Communications*. 2008;15(5):80-85
- [5] Asante M, Akomea-Agyin K. Analysis of security vulnerabilities in Wifi-protected access pre-shared key (WPA-PSK/ WPA2-PSK). *International Research Journal of Engineering and Technology (IRJET)*. 2019;06(01):537-545
- [6] Singh R, Sharma TPA. Location based method for restricting the flooding DoS effect in WLANs. *Journal of Location Based Services*. 2016;9(4):273-295. Taylor and Francis
- [7] Singh R, Sharma TP. A key refreshing technique to reduce 4-way handshake latency in 802.11i based networks. In: *Proceedings of the Fourth IEEE International Conference on Computer and Communication Technologies, ICCCT'13*; September 2013; Allahabad. pp. 157-163
- [8] Singh R, Sharma TP. A sequence number based WLAN authentication scheme for reducing the MIC field overhead. In: *Proceedings of the Tenth IEEE International Conference on Computer and Communication Technologies, WOCN'13*; July 2013; Bhopal. pp. 1-4
- [9] Newman LH. The Secure Wi-Fi Standard Has a Huge Dangerous Flaw [Internet]. 2017. Available from: <https://www.wired.com/story/krack-wi-fi-wpa2-vulnerability/>
- [10] Wi-Fi Alliance. Discover Wi-Fi Security [Internet]. Available from: <https://www.wi-fi.org/discover-wi-fi/security>
- [11] Li X, Bao F, Li S, Ma J. FLAP: An efficient WLAN initial access authentication protocol. *IEEE Transactions on Parallel and Distributed Systems*. 2013;25(2):488-497
- [12] Martinovic I, Zdarsky FA, Bachorek A, Schmitt JB. Measurement and analysis of handover latencies in IEEE 802.11i secured networks. In: *Proceedings of the European Wireless Conference (EW2007)*; April 2007; Paris. pp. 1-7
- [13] Singh R, Sharma TPA. Secure WLAN authentication scheme. *IEEK (Institute of Electronics Engineers of Korea) Transactions on Smart Processing and Computing*. 2013;2(3):176-187
- [14] Tang C, Wu DO. An efficient mobile authentication for wireless networks. *IEEE Transactions on Wireless Communications*. 2008;7(4):1408-1416
- [15] Park CS. Two-way handshake protocol for improved security in IEEE 802.11 wireless LANs. *Computer Communications*. 2010;33(9):1133-1140
- [16] Singh R, Sharma TP. A key hiding communication scheme for enhancing the wireless LAN security. *Springer Wireless Personal Communications*. 2014;77(2):1145-1165

- [17] Singh R, Sharma TP. Simulated analysis of a cryptographic solution for WLANs against Denial of Service (DoS) attacks. *Journal of Engineering Science and Technology*. 2014;**9**(Special Issue):57-67
- [18] Singh R, Sharma TP. Modeling and performance evaluation of computational DoS attack on an access point in wireless LANs. In: Ram M, Davim JP, editors. *Advanced Mathematical Techniques in Science and Engineering*. Denmark and The Netherlands: River Publishers; 2018. pp. 101-120
- [19] Johnson H, Nilsson A, Fu J, Wu SF, Chen A, Huang H. SOLA: A one-bit identity authentication protocol for access control in IEEE 802.11. In: *Proceedings of the IEEE Global Telecommunications Conference*. Taipei, Taiwan; Vol. 1; 17-21 November 2002. pp. 768-772
- [20] Wu F, Johnson H, Nilson A. SOLA: Lightweight security for access control in IEEE 802.11. In: *Wireless Security*. IEEE IT Professional; 2004;**6**(3):10-16
- [21] Wang H, Velayutham A, Guan Y. A lightweight authentication protocol for access control in IEEE 802.11. In: *Proceedings of the IEEE Global Telecommunications Conference, GLOBECOM'03*. San Francisco, CA, USA; 2003. pp. 1384-1388
- [22] Wang H, Cardo J, Guan Y. Shepherd: A lightweight statistical authentication protocol for access control in wireless LANs. *Computer Communications*. 2005;**28**(14):1618-1630
- [23] Ren K, Lee H, Han K, Park J, Kim K. An enhanced lightweight authentication protocol for access control in wireless LANs. In: *Proceedings of the 12th IEEE International Conference on Networks*. Singapore; Vol. 2; 2004. pp. 444-450
- [24] Lee Y-S, Chien H-T, Tsai W-N. Using random bit authentication to defend IEEE 802.11 DoS attacks. *Journal of Information Science and Engineering*. 2009;**25**(5):1485-1500
- [25] Pepyne DL, Ho Y-C, Zheng Q. SPRiNG: Synchronized random numbers for wireless security. In: *Proceedings of the IEEE Wireless Communications and Networking, WCNC'03*. New Orleans, LA, USA; 2003. pp. 2027-2032
- [26] Singh R, Sharma TP. A novel sequence number based secure authentication scheme for wireless LANs. *Journal of Electronics Science & Technology (JEST)*. 2015;**13**(2):144-152
- [27] Juang W-S, Wu J-L. Two efficient two-factor authenticated key exchange protocols in public wireless LANs. *Computers and Electrical Engineering*. 2009;**35**(1):33-40
- [28] Lee Y, Kim S, Won D. Enhancement of two-factor authenticated key exchange protocols in public LANs. *Computers and Electrical Engineering*. 2010;**36**(1):213-223
- [29] Lei J, Fu X, Hogrefe D, Tan J. Comparative studies on authentication and key exchange methods for wireless LAN. *Computers & Security*. 2007;**26**:401-409
- [30] Lee I, Hunt R. A novel design and implementation of DoS-resistant authentication and seamless handoff scheme for enterprise WLANs. In: *Proceedings of the 8th Australian Information Security Management Conference*; Perth, Australia; 2010. pp. 49-61
- [31] Kassab M, Belghith A, Bonnin J-M, Sassi S. Fast pre-authentication based on proactive key distribution for 802.11 infrastructure networks. In: *Proceedings of the Wireless Multimedia*

Networking and Performance Modeling,
WMuNeP'05. Montreal, Quebec,
Canada; 13 October 2005. pp. 46-53

[32] Singh R, Sharma TP. Secure WLAN
handoff scheme with continuous
authentication. MIS Review.
2016;**21**(1):35-50

[33] Jiang X, Liu M, Yang C,
Liu Y, Wang R. A blockchain-based
authentication protocol for WLAN mesh
security access. CMC. 2019;**58**(1):45-59

Analysis of Network Protocols: The Ability of Concealing the Information

Anton Noskov

Abstract

In this chapter, we consider the possibility of hidden data. Since today all network services rely on the basic protocols, the use of untestable and redundant fields may become a big problem. All of the modern data protocols have vulnerabilities. An attacker can use the reserved fields or field use undocumented way. Depending on the data transmission method and detection mechanisms, the technology for assessing the possibility of transmitting hidden information is changing. The work is of great practical interest for the implementation of systems to detect and prevent intrusions and data leaks in it. The authors determine the possibility of transmission and detection sends using a comparative evaluation of the fields in the packet with the values recommended in the standard protocol.

Keywords: network protocols, transport protocols, network analyze, network security

1. Introduction

Network steganography—type of steganography, in which secret data carriers use the network protocols of the OSI reference model—the open systems interconnection network model. In general, network steganography is a family of methods for modifying data in the headers of network protocols and in the payload fields of packets, changing the structure of packet transmission and hybrid methods in a particular network protocol (and sometimes several at once).

The transfer of hidden data in network steganography is carried out through hidden channels. The term “covert channel” introduced by Simmons in 1983 determined that the problem of information leakage is not limited to the use of software. A covert channel can exist in any open channel in which there is some redundancy. The hidden data is called steganogram. They are located in a specific carrier (carrier).

In network steganography, the role of the carrier is carried out by the packet transmitted over the network. The main parameters of network steganography are the bandwidth, covert channel, probability of detection, and steganographic cost. Bandwidth is the amount of secret data that can be sent per unit of time. The probability of detection is determined by the possibility of detecting a steganogram in a particular carrier. The most popular way to detect a steganogram is to analyze the statistical properties of the data obtained and compare them with typical values for this carrier. Steganographic cost characterizes the degree of change in the carrier after exposure to the steganographic method.

1.1 Network steganography methods

Baseline data for consideration classifications of methods and means of network steganography come from the materials of Polish scientists Mazurczyk and Szczypiorski and reports on the experiments of Canadian scientists Ahsan and Kundur, scientists Cauch and Gomez of the University of California at Irvine, and researchers Handel and Sandford at the National laboratory at Los Amos. All materials are freely available. Network steganography methods can be divided into three groups [1]:

- Steganography methods, whose essence is in changing data in the fields of the network protocol headers and in the packets payload fields.
- Steganography methods, in which the structure of packet transmission changes, for example, the sequence of packet transmission or the intentional introduction of packet loss during transmission.
- Mixed (hybrid) methods of steganography—when they are used, the contents of the packages, the delivery times of the packages, and the order of their transfer change.

Each of these methods is divided into several groups; for example, package modification methods include three different methods:

- Methods for changing data in protocol header fields: they are based on modifying the IP, Transmission Control Protocol (TCP), SCTP header fields, and so on.
- Packet payload modification methods; in this case, various watermark algorithms, speech codecs, and other steganographic techniques for hiding data are used.
- Methods of mixed techniques.

Methods for modifying the structure of gears and packages include three guidelines:

- Methods in which the order of the sequence of packets is changed.
- Methods that change the delay between packets.
- Methods, the essence of which is to introduce intentional packet loss by skipping sequence numbers at the sender.

Mixed (hybrid) methods of steganography use two approaches: methods of audio packet loss (LACK) [2] and packet retransmission (RSTEG) [1].

The main idea of methods for modifying header fields is to use some header fields to add steganogram to them [3, 4]. This is possible due to some redundancy in these fields, that is, there are certain conditions in which the values in these fields will not be used in the transmission of packets. The most commonly used header fields are IP and TCP protocols.

Consider an example of a similar method based on modifying unused IP protocol fields to create a hidden channel [4].

The value of the “Identification” field of the IP packet is generated to the sender side. This number contains a random number that is generated when a package

is created. The “Identification” field is used only when fragmentation is used. Therefore, to use this method, you need to know the MTU value in the transmitted network and not exceed it, so that the packet is not fragmented during transmission. In the absence of the need for packet fragmentation, a certain redundancy occurs in the “Flags” field, in the second bit, which is responsible for setting the Don’t Fragment (DF) flag. It is possible to specify a flag notifying the sender’s unwillingness to fragment a packet. If the steganogram package is not fragmented due to its size, you can hide the information in the “DoNotFragmentBit” flag field. Using this method provides bandwidth of 1 bit.

The advantage of this method is the transmission of unchanged information from the sender to the recipient, but it also limits the amount of information sent. Steganography based on this method is easily implemented; has a good bandwidth, since you can send a lot of IP packets with the changes; and is low cost due to the use of fields that do not violate the functionality of the packet. Among the shortcomings it should be noted that the transmitted data is contained in the open form and can be easily read by the observer (although it is possible to strengthen the protection using additional cryptography).

Another method of modifying network packets that alters the payload of a VoIP packet can be widely used in practice with the popularity of programs that provide voice and video communications over the Internet. The network steganography method designed to hide VoIP messages is called Transcoding Steganography (TranSteg), a network steganography method that compresses the payload of a network packet by transcoding. TranSteg can be used in other applications or services (e.g., streaming video), where there is a possibility of compression (with or without losses) of open data. In TranSteg, data compression is used to make room for the steganogram: transcoding (lossy compression) of voice data from a high bitrate to a lower bitrate occurs with minimal loss of voice quality, and after compression, data is added to the free space in the payload package [5]. In general, the method allows to obtain more or less good steganographic bandwidth of 32 kb/s with the smallest difference in packet delay. Experiments of Polish scientists have shown that the delay in transmitting a VoIP packet using TranSteg increases by 1 ms, in contrast to a packet without a steganogram. The complexity of detection directly depends on the choice of the scenario and the conditions of the outside observer (e.g., its location). Among the shortcomings worth mentioning is the fact that this method is difficult to implement. It is necessary to find out which codecs the program uses for voice communication, to choose codecs with the smallest difference in speech quality, while giving more space for embedding steganograms. During compression, the quality of the transmitted speech information is lost.

Also interesting is the direction using the mechanisms of the SCTP protocol. Stream control transport protocol (SCTP) [6] is a packet-based transport protocol, a new-level transport protocol that will replace TCP and User Datagram Protocol (UDP) in future networks. Today, this protocol is implemented in operating systems such as BSD, Linux, HP-UX, and SunSolaris, supports network devices of the Cisco IOS operating system, and can be used in Windows. SCTP steganography uses new features of this protocol, such as multi-threading and the use of multiple interfaces (multi-homing).

The methods of SCTP steganography can be divided into three groups [7]:

- Methods in which the contents of SCTP packets change.
- Methods in which the sequence of transmission of SCTP packets is changed.
- Methods that affect both the content of packages and their order when transfer (hybrid method).

Methods for changing the contents of SCTP packets are based on the fact that each SCTP packet is made up of parts and each of these parts can contain variable parameters. Regardless of the implementation, a statistical analysis of the addresses of the network cards used for the forwarded blocks can help in detecting hidden connections. Eliminating the possibility of applying this method, steganography can be achieved by changing the source and destination addresses in randomly selected packet, which is contained in the re-expect PTO unit.

The essence of the hybrid method based on the SCTP protocol is to use certain protocol mechanisms that allow you to organize the intentional passing of packets in a stream without resending it. Later a steganogram is added to this packet, and it is resubmitted [7]. Modification of packages using a hybrid method can be presented on the Hidden Communication System for Corrupted Networks (HICCUPS), which uses the imperfections of data transmission in a network environment, such as interference and noise in a communication environment, as well as the usual susceptibility of data to distortion. HICCUPS is a steganographic system with bandwidth allocation in a public network environment. Wireless networks are more susceptible to data corruption than wired ones, so the use of noise and noise in the communication environment during system operation looks very tempting. "Listening" of all the frames with the transmitted data in the environment and the ability to send damaged frames with incorrectly corrected code values are two important network features necessary for the implementation of HICCUPS. In particular, wireless networks use an air connection with a variable bit error rate (BER), which makes it possible to introduce artificially damaged frames. This method has low bandwidth (network dependent), cumbersome implementation, low steganographic cost, and high detection complexity. However, the frame analysis does not involve checksum may lead to the discovery of the use of Nogo given method.

The RSTEG method is based on the packet resending mechanism, the essence of which is as follows: when the sender sends a packet, the recipient does not respond with a confirmation flag; thus the packet resending mechanism should work, and the packet with the steganogram inside will be sent again, but confirmation does not come. The next time this mechanism is triggered, the original packet is sent without hidden attachments, to which the packet arrives with confirmation of successful receipt.

The performance of an RSTEG depends on many factors, such as the details of the communication procedures (in particular, the size of the packet payload, the frequency with which segments are generated, and so on).

The investigated method of steganography using packet retransmission RSTEG is a hybrid. Therefore, its steganographic bandwidth is approximately equal to the bandwidth of the methods with packet modification and at the same time higher than the methods of changing the order of packet transmission. The complexity of detection and throughput is directly related to the use of the implementation mechanism of the method. RSTEG based on RTO is characterized by high detection complexity and low bandwidth, while SACK has the maximum bandwidth for RSTEG, but is also more easily detected. The use RSTEG utilizing TCP protocol is a good choice for IP networks. Among the shortcomings, it should be noted that this method is difficult to implement, especially its scenarios, which are based on interception and correction of packets transmitted by ordinary users. Due to the dramatically increased frequency of retransmitted packets or the unusual occurrence of delays in the transmission of steganograms, a casual observer may be suspicious.

Lost audio packets steganography (LACK)—steganography of deliberate delay of audio packets [2]. This is another method implemented via VoIP. Communication over IP telephony consists of two parts: signaling (dialing) and conversational. Both parts of the traffic are transmitted in both directions. The signaling protocols used are SIP and RTP (with RTCP acting as the control protocol). This means that during the signaling phase of the call, the SIP endpoints (called user SIP agents) exchange some SIP messages. Usually SIP messages pass through SIP servers: proxy or redirected, which allows users to search and find each other. After this stage, the conversation phase begins, where the audio (RTP) stream goes to both directions between the caller and the callee. This method has certain advantages. The bandwidth is not less and sometimes higher than the other algorithms that use audio packets. But if you intentionally cause losses, the quality of the connection deteriorates, which can become suspicious for both ordinary users and listeners. Based on the presented steganalysis LACK methods, it can be concluded that the method has an average detection complexity. The implementation of the method is too complex, but may not be possible within certain operating systems.

Table 1 shows a comparison of methods and their main characteristics and implementation. The position of each method in this table shows how much its characteristics are superior or inferior to the others. The higher the method displayed at the table, the more indicators of its characteristics. In the “Implementation” field, the simplicity of the organization of this method is considered. The less time and effort required by the implementation of this method, the higher its position in this title. Based on the data from **Table 1**, it can be concluded that the main characteristics are directly dependent on each other.

No	Throughput ability steganography	Complexity discoveries	Steganography cost	Implementation
1	TranSteg	HICCUPS	HICCUPS	Modification header fields TCP and IP packets
2	LACK	TranSteg	LACK	Modification data blocks in SCTP protocols
3	HICCUPS	LACK	RSTEG	TranSteg
4	RSTEG	RSTEG	TranSteg	Using SCTP multi-homing
5	Modification fields in TCP headers and IP packets	Using SCTP protocol (hybrid)	Protocol use SCTP (hybrid)	Using SCTP protocol (hybrid)
6	Modification data blocks in SCTP protocols	SCTP multi-homing	Modification of blocks data in SCTP protocols	LACK
7	Using SCTP protocol (hybrid)	Modification fields in TCP headers and IP packets	SCTP multi-homing	RSTEG
8	Using SCTP multi-homing	Modification data blocks in SCTP protocols	Modifying fields in TCP and IP headers packages	HICCUPS

Table 1.
Comparison of network steganography methods.

2. The combined method using modification of the fields IP and TCP

As mentioned earlier, the methods for modifying the IP and TCP header fields have certain features that make them stand out from the rest of the methods:

- The most common and standard protocols are used as carriers of the steganogram.
- Total gives bandwidth of 49 bits per 1 packet.
- Implemented on any operating system, the implementation does not require long adjustments and preparations.
- Changes in the package will not affect its behavior on the network, in case it will not be fragmented.

Despite the many advantages of both methods, there are some flaws, and the main one, to which attention is immediately drawn, is the obviousness of data transfer, i.e., any statistical analysis allows us to calculate both the hidden communication channel itself and the information transmitted in it.

The method proposed by Rowland [3] is as follows: to generate a value in the “Sequence Number” field, the plaintext character is encoded in accordance with the ASCII table, and the resulting value is multiplied by a certain number multiple of two. The resulting value is entered in the “Sequence Number” field and sent to the recipient. The recipient, knowing the key (divisor), should check all incoming TCP packets for the subject of the steganogram, dividing the value of the “Sequence number” field by the key.

On the one hand, this method allows you to create a data channel through which you can transmit secret data in front of a passive observer. But the existence of a single key is a disadvantage, since, based on a dozen of such packages, it can be concluded that the sequence numbers of all packages have a common factor, which is the key. Thus, the proposed method is easy to detect.

Based on the source data and analysis of the disadvantages of network steganography methods with modification of the IP and TCP packet header fields, we can propose a modified method that will be based on the simultaneous use of the IP and TCP protocol header fields. The key needed to decrypt the transmitted message will also be transmitted as a steganogram, only in encrypted form in the “Identifier” field of the IP header, while the encrypted steganogram will be transmitted in the “Sequence number” field of the TCP header.

The implementation of this method is divided into two parts:

- Preparing data for the transfer, which includes generating the key k , converting the transmitted secret symbol or number into its corresponding code in the ASCII table, and calculating the value of the carrier C , which is an encrypted steganogram.
- Entering data into the corresponding TCP and IP header fields.

The first block consists of the following steps:

- Generation of the key k , which will be used in the future. The key can be any number that is a multiple of two. To generate a key, take two numbers x and y and raise the first to the power of the second.

- The conversion of secret data—a character or number that must be transferred to the corresponding code in the ASCII table. The coded number is denoted by S, since it is our steganogram.
- Getting the media C as the product of the key value by the value of a secret character.

$$C = S * k_{10}$$

- Checking the number C—it must meet the requirement $2^{28} < C < 2^{33}$. This condition is necessary so that the value of the “Sequence number” field does not look suspicious. If the value of C does not meet the requirements, the numbers x and y need to be changed to others, and repeat steps 1–2. Further studies will be conducted on the automatic formation of x, y.
- The value of the numbers x and y is written together into the number z and is flipped so that the previous values can only be read from right to left.

Then the data is converted from decimal to hexadecimal. Thus, we get a three-digit hexadecimal number inv. (z) 16.

Then, at the second stage, you need to put the obtained values of the encrypted key and steganogram into the TCP and IP header fields.

We briefly describe the network steganography method with a modification of the fields in the TCP header, since in it we will transmit the secret message itself. For the purpose of steganography, the header of this protocol usually uses some fields that can be changed without losing the functionality of the package. For the purpose of our research, we will focus on the “Sequence Number” field (SN, SequenceNumber). This field performs two tasks. The first is the following: if the SYN flag is set, then this initial value of the sequence number is ISN (InitialSequenceNumber), and the first byte of data that will be transmitted in the next packet will have a sequence number equal to ISN + 1. Otherwise, if SYN is not set, the first byte of data transmitted in this packet has this sequence number. For our case it is important to know that this value will not change during the path of the packet from the sender to the recipient.

The “Sequence Number” field allows you to create a 32-bit length sequence. According to the Rowland method, the transmitted message is encoded in accordance with the ASCII table and multiplied by a certain number (the key), a multiple of two to reduce the detection probability, then entered into the generated TCP packet in the “Sequence number” field, and the packet is sent. When the packet reaches the destination address, the recipient must save all incoming TCP packets, from which he must remove the value in the “Sequence number” field and then divide by the key he knows in advance. But, as it was said before, this method is extremely easy to detect based on the analysis of a number of TCP packets due to a permanent key. In the proposed modification of the method, this key will be transmitted simultaneously with the TCP packet, in the IP header. This will increase the difficulty of detecting the steganogram.

The next step is to add the value of the C media in the “Sequence Number” field of the TCP header.

Next, you must enter the value of the encrypted key (inv (z)) 16 in the IP header field. To organize such an operation, you should return to the network steganography method with modification of the IP header fields. During the packet path, only the “Identifier” field remains unchanged; its length is 16 bits and 1 bit in the “Flags” field, which is responsible for the DF flag. Changing these fields does not carry

changes in the package, in case the package is not fragmented, but it should not be, since by condition we need to know the minimum MTU value and not exceed it when creating and sending the package.

At the “Identifier” field, 16 bits is available to us for adding a steganogram; the information in it is displayed in the form of four numbers in hexadecimal number system. Thus, we have 65,535 possible values that can be used both for transmitting the steganogram and for the key, which in turn is also a steganogram. In order not to transmit the key in such an explicit form, it is proposed to use only three numbers out of four, while reading them from right to left. In this case, the number can be odd with its standard reading from left to right. The fourth unused number can take any value. Thus, we can use only 16 of the 17 bits available in a packet. It is proposed to use the second bit in the “Flags” field—DF—as a specific label, the presence of which allows you to expand the key extraction algorithm: whether you need to read the value from the first or from the second number in the “Identifier” field to extract the key.

Thus, the next step is to enter (inv (z)) 16 in the “Identifier” field of the IP header. At the same time, we must set the value of “1” to the second bit in the “Flags” field if we enter the key in the first 12 bytes of the “Identifier” field or 0 if we fill the first 4 bytes of the field with random values and in the remaining 12 bytes our key.

Next, we send a packet with modified fields to the recipient, where he must carry out the procedure inversely described in the framework of this algorithm [8].

We calculate the bandwidth of the proposed method.

Since the “Identifier” field in the IP header can contain 16 bits of information, 1 bit is available in the “Flags” field, and in the “Sequence number” field, a 32-bit information is available in the TCP header; we can conclude that the total throughput of steganography is 49 bits. But it should be noted that in this method we use the “Identifier” field to transmit the encrypted key in the steganogram, which is used to extract secret information from the “Sequence number” field, and the bit in the “Flags” field is used as a label. Thus, to transfer the encrypted key, we allocate 12 bits of information available in the “Identifier” field, and in the remaining 4 bits, we enter a random number from 0 to 16 in the hexadecimal number system (from 1 to F) and use 1 bit as a label, necessary for more organization more flexible operation of the algorithm. Based on this, we can conclude that for transmitting specific information, we have 32 bits left in the “Sequence number” field, and 3 bits of secret information can be transmitted, which is encrypted in 32 bits of information hiding the secret.

2.1 Intercomputer exchange

The exchange of computer networks is based on the Open System Interconnection (OSI) reference model.

Studying hidden information flows with computer interaction on networks of interest will include information about the services that are added to the network traffic data. As part of the protocol, headings are assessed at two levels: network and transport. We will address network protocols (IPv4 and IPv6) and transport protocols (TCP and UDP).

Further, we are considering the reports and the possibility of more detailed manipulation.

2.2 IPv4

IPv4 is the most popular protocol of network level; see more information in RFC791.

The header size of IPv4 is 20 bytes; using specialized field in header—“Options” field—can increase it. When the amount of the header is less than 20 bytes, it is likely damaged and has to be discarded.

2.3 Header of IPv4

The format of IPv4 header is presented in **Figure 1**.
 IPv4 header field analysis shows the following results:

1. “Internet Header Length” field. Ability to increase the size of the Internet Header Length field to extend the original header. This change allows you to add data to the next two “Options” and “Padding” fields.

2. “Type of Service” field

Bits from 0 to 2 are set for priority and 6 to 7 set to reserved.

-0-2:

The value “111” should not appear on the networks of provider; it could be appearing only for local networks, which leads to the point that the capture of this value in the network provider is a mark of malicious information injection.

-6-7:

By default, these bits are reserved and must be set to 0; the result is that the other value is possible injection information.

3. “Identification” field

You can change the value of the identification field. The point is that the field is used to build correctly after fragmentation, but there is a DF flag that rejects fragment packets, so if the flag is set to “1” this ID is not required, and this field could be used to pass hidden information.

4. “Flags” field

As the standard requires, the first bit is reserved and should be set to “0”; if the result is different, it is mark of injection information.

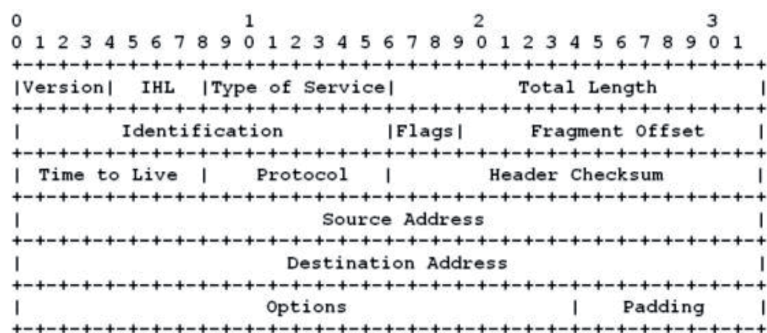


Figure 1.
 Header of IPv4.

5. “Fragment Offset” field

You can change the value of the “Fragment Offset” field. The best option is when the DF flag is set to “0,” since the fragmentation strategy is designed so that an unfragmented datagram in all fields related to fragmentation has zero values. This means, despite the fact that the flag prevents fragmentation, we can still implement it in the offset of the fragment, but the fact of identification of the manipulation becomes more detectable.

6. “Source Address” field

You can change the “Source Address” field value.

7. It should be noted that manipulation is possible only on the condition that the package consists of hidden source data. Since the manipulation will not be caused by the source of the information, the receiving site could not properly build the packets.

8. “Destination Address” field

IPv4-in IPv6 headers can be encapsulated using the IPv4 Destination Address field to insert information into it. In this case, the IPv6 header will be responsible for delivering the package.

9. “Options” field

The value of the options field is limited in the IPv4 header, and as a result of the analysis, we are trying to determine any field value that may appear in this type of field. So we may try to determine the incorrect significant of this field, the appearance of which indicates the possible malicious activity on the injection of information.

10. “Padding” field

This field goes after value 0x00 of the “Options” field; the value is the EOL and takes up to 32-bit header boundaries. The interest in this manipulation is that after the optional EOL, the equipment does not examine headers on 32-bit boundaries; this means that these bytes are invisible to network devices and sniffer. Although the analysis of this field is simple enough, the EOL up to 32-bit header boundaries must be set to “0” at the standard behind the “Options” field, causing any other value of this field to indicate that the data is being injected.

2.4 Injection’s result

The standard IPv4 header size with options and fields with padding is 320 bits. Two different options need to be considered:

1. IPv4 is a carrier and is responsible for packet addressing. Due to manipulation, 182 bits can be used, which is 56.88% of the total number of bits. This volume allows you to insert 22 symbols from 8 bits in ASCII encoding into the header.

So after calculations we have got a value up to 4 bits. This remainder is part of the other 8 bits of the transmitted information.

2. IPv4 is a passenger, it's an IPv4 encapsulated header in other headers, such as IPv6 or GRE. In this case, the method for implementing the target address can be used. As a result, handling bits 214, 66% of the total number of bits can be used. This volume allows you to implement a 26-character header with 8 bits in ASCII encoding. Thus, after calculations, a value of 6 bits is obtained. The treated residue was included in an additional 8 bits of the transmitted symbol.

2.5 IPv6

2.5.1 Header of IPv6

The header's format of IPv6 is presented in **Figure 2**.

1. "Traffic Class" field

You can change the "Traffic Class" value arbitrarily. This manipulation cannot be detected by analysis.

2. "Flow description" field

You can change the value of the "Flow Label" field.

This manipulation cannot be detected by the packet sniffer.

3. "Load Length" field

It is possible to increase the size of this field when adding data to the end of the original IP packet, like IPv4. This modification cannot be detected by the packet sniffer.

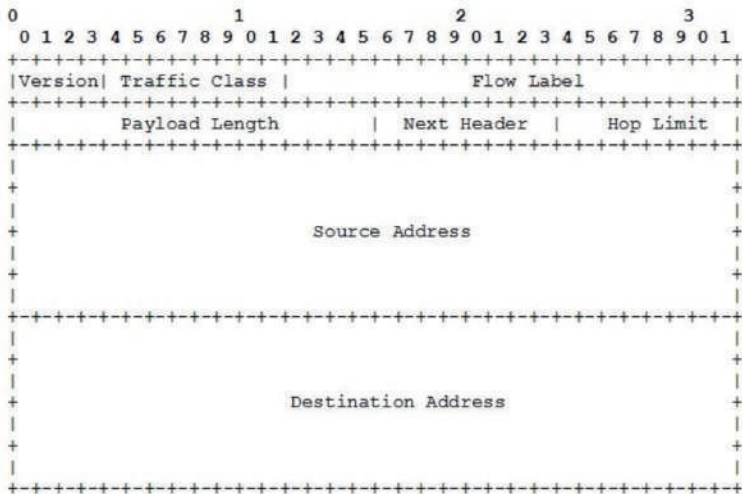


Figure 2.
Header format of IPv6.

4. “Source Address” field

You have the possibility to change the data of this field at IPv4 format, but international standards from the IPv6 community do not recommend using it as a source address.

5. “Destination Address” field

In this protocol, you can use the IPv6 “Destination Address” field in the IPv4 encapsulation header to load information into it. In this case, the IPv4 header will be responsible for the packet delivery.

This manipulation cannot be detected by the packet sniffer.

3. Result of injection

The standard IPv6 header size with options and fields with padding is 320 bits. Two different options need to be considered:

1. IPv6 is a carrier, that is, it is responsible for addressing the package. As a result of the manipulations described above, 156 bits can be used, which is 48.75% of the total number of bits. This volume allows you to insert a caption with 19 characters from 8 bits into the ASCII character set. Thus, after calculations get a value of 4 bits. The treated residue was included in an additional 8 bits of the transmitted symbol.
2. IPv6 is a passenger and is transmitted by IPv6 encapsulation header to other headers, such as IPv4 or GRE. In this case, the method for implementing the target address can be used. As a result of the manipulations described above, it is possible to use 284 bits, which is 88.75% of the total number of bits. This volume allows you to implement a 35-character header with 8 bits in ASCII. Thus, after calculations, we get a possible value of 4 bits. The processed remainder will be added as an additional 8 bits of transmitted characters.

3.1 TCP

Transmission Control Protocol is a reliable protocol of transport layer. TCP is oriented to establish a logical connection, that is, the hosts negotiate and create a session and then begin to transfer data. Every time a package is sent, the sender is awaiting acknowledgement of delivery receipt. This protocol is standardized by RFC 793.

3.1.1 Header of TCP

Header's format of TCP is presented in **Figure 3**.
“Source Port” field

1. You can change the “Source Port” field value. Processing is only possible when the package was a hidden data source. Due to the manipulations that occur on the source host, the receiving party will not be able to properly assemble the original packet. This manipulation cannot be detected by the packet sniffer.

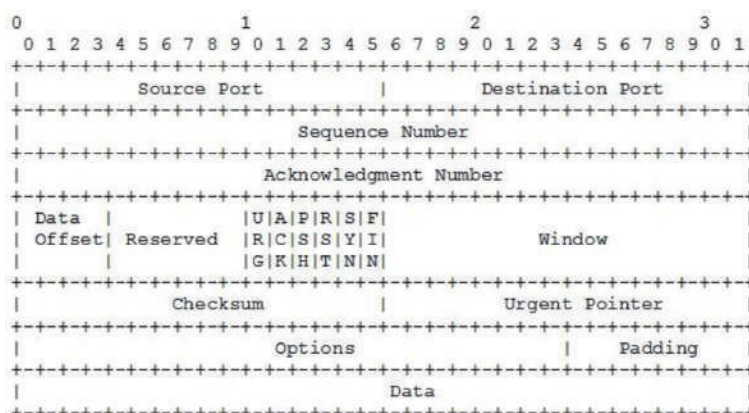


Figure 3.
 Header format TCP.

2. “Destination Port” field

You can change the value of “Destination Port” field. Handling is only possible when the packet was a hidden data source. Due to the manipulations that occur on the source host, the receiving party will not be able to properly assemble the original packet. This modification cannot be detected by sniffer.

3. “Sequence Number” field

We could modify the information in this field. Processing is only possible when the package was a source of hidden data. Because of the modification that had occurred at the source device, the receiving PC cannot correctly build the original packets. This modification cannot be observed by the network sniffer.

4. “Acknowledgment Number” field

We could change the contents of this field. Modification is allowed provided that the package was made up source of hidden data. Due to the modification that occurred on the source host, the receiving party will not be able to properly assemble the original packet.

5. “Data Offset” field

The manipulation is as follows: this increases the size of the “Data Offset” field, expands the TCP header, and adds a parameter field. In the options you can add data after byte 0x00 EOL.

At standard byte 0x00 EOL, bytes with a value of “0” should be due to some other value that indicates that a data injection has occurred.

6. “Reserved” field

You can modify the value of this field.

By default, the values of all standard bits must be set to “0” as a result of some other values that indicate that a data injection has been occurred.

7. “Window” field

You can modify the value of the “Window” field. Handling is only possible when the packet was built at a hidden data source. Due to the manipulations that occur on the source host, the receiving party will not be able to properly assemble the original packet

8. “Pointer Urgent” field

You can modify the value of this field. This injection is only possible if all URG options are present.

So, if the Urgent Pointer is filled in and the flag of URG is not setting, it means that the Urgent Pointer is not used correctly.

9. “Options” field

We could modify the data of this field. In the options, you can realize the data after value 0x00, but it is not considered after this byte header data.

TCP header option values are limited, and network analysis results in attempting to identify a possible option that attempts to identify incorrectly filled options or unknown options whose appearance indicates a possible injection of information.

10. “Padding” field

It is possible to fill the field of any padding.

It should be noted that manipulation is only possible if the package is made up of hidden source data. Because of the manipulation that occurs at the source, the receiving party cannot properly collect packets.

Handling “Padding” is one of the most interesting. The “Padding” field starts after the 0x00 in the “Options” field; the value is the EOL option and takes up to 32-bit header boundaries. Interest in this manipulation is contained in the following text after the EOL does not produce a 32-bit header, which means that these bytes are invisible to network devices and sniffer. Although the analysis of this field is simple enough, the EOL up to 32-bit header boundaries must be set to “0” at the standard behind the “Options” field, causing any other value of this field to indicate that the data is being injected.

4. Result of injection

The standard TCP header field with options and fall is 192 bits. As a result of the above actions, you can use up to 150 bits, which is 78.13% of the total number of bits in the original, unmodified header. This amount of data allows the use of 18 characters in an 8-bit header in the standard ASCII character set. Therefore, after all the calculations, we get the maximum possible amount equal to 6 bits. The processed piece of information was included in the next 8 bits of the transmitted symbol.

4.1 UDP

User Datagram Protocol is a connectionless transport layer protocol. No connection setup is created before transferring between hosts. This protocol is less reliable than TCP, but gives a higher transfer rate with less overhead. This protocol is standardized by RFC 768.

4.1.1 Header field of UDP

Header's format of UDP is presented in **Figure 4**.
 "Source Port" field

1. You can change the "Source Port" field value. Processing is only possible when the package was a hidden data source. Because of the manipulation that had occurred at the source device, the receiving party cannot properly assemble the original packets. This manipulation cannot be detected by the packet sniffer.

2. "Destination Port" field

You can change the "Target Port" field value. Processing is only possible when the package was created by a hidden data source. Due to the manipulation of the device generating the packages, the receiving party cannot correctly assemble the source packages. This modification cannot be detected by the network sniffer.

3. "Length" field

We could change the significance of the "Length" field. Increasing the value of this field has also increased the size of the package, so we can change the fields of data octets by appending to the end of datagram.

So processing is possible when the package was a source of hidden data. So if the modification had occurred at the source device, the receiving host cannot properly assemble the original packets. This manipulation cannot be detected by the packet sniffer.

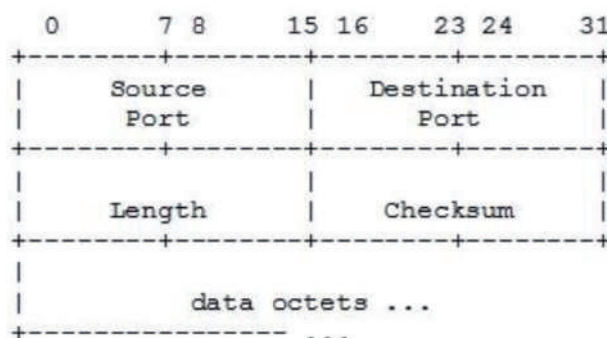


Figure 4.
 Header of UDP.

5. Result of injection

The size of the UDP header of the datagram is 64 bits; as a result of the described changes, you can use 32 bits, which is 50% of the total number of bits in the header, which allows you to implement a 4–8-bit header in the ASCII character set.

6. Conclusion

In this work, we began to develop methods and special software for generating bitstreams in order to organize a secure connection.

This software method was implemented in software, ensuring secure network communication. The main part of the program model is a detector program for analyzing network traffic to search for possible hidden transmissions. The analysis is implemented by checking the header in compliance with the standards, which is needed to identify unauthorized values for specific areas of the PDU.

The surveys revealed possible vulnerabilities that could embed relevant information in the puncture headers we reviewed. **Table 2** presents the quantification of the study results, showing the remainder is the number of bits that are part of the next 8 bits of the transmitted symbol.

It should be noted that TCP was created as a reliable protocol for delivery, but after entering the hidden data by the TCP header proposed above, changes made to the header fields of the TCP lead to the loss of the functionality of a reliable protocol, making it similar to the UDP.

In the created model, the transmission of one packet is realized, that is, the full message is embedded in all possible of headers at only one datagram. In order to see the maximum feasible messaging, we chose the following protocols: IPv4, IPv6, and TCP. Note that for simplicity, TCP header data is not included in the fragment offset. Thus, thanks to the proposed manipulation, the programming model uses 603 bits, which is 74.04% of the total number of bits in the order of three headers. This volume allows you to enter 75 characters out of 8 bits in ASCII encoding.

Protocol	Size of injection information (bits)	Percentage of the total header size (%)	The number of symbols	Rest bits
IPv4 (carrier)	182	56.88	22	6
IPv4 (passenger)	214	66	26	6
IPv6 (carrier)	156	48.75	19	4
IPv6 (passenger)	284	88.75	35	4
TCP	150	78.13	18	6
UDP	32	50	4	0

Table 2.
The quantification of the study results.

Author details

Anton Noskov
Yaroslavl State Technical University, Yaroslavl, Russia

*Address all correspondence to: anton.noskov@gmail.com

IntechOpen

© 2020 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



References

- [1] Mazurczyk W, Szczypiorski K. In: Meersman R, Tari Z, editors. *Steganography of VoIP Streams*. Springer-Verlag; 2009. Available from: http://home.elka.pw.edu.pl/~wmazurcz/moja/art/OTM_StegVoIP_2008.pdf
- [2] Mazurczyk W, Szaga P, Szczypiorski K. *Retransmission Steganography and Its Detection*. Available from: <http://cygnus.tele.pw.edu.pl/~wmazurczyk/art/RSTEG.pdf>
- [3] Rowland CH. Covert channels in the TCP/IP protocol suite. *Central European Journal of Computer Science*. 1997;2(5):45-66. Available from: <http://firstmonday.org/htbin/cgiwrap/bin/ojs/index.php/fm/article/viewArticle/528/449>
- [4] Cauich E, Gómez R, Watanabe R. *Data Hiding in Identification and Offset IP fields*. California University at Irwing, Computer Science and Engineering, Irvine, CA, USA: University of California. <http://www.sciweavers.org/read/data-hiding-in-identification-and-offset-ip-fields-124683>
- [5] Mazurczyk W, Szaga P, Szczypiorski K. *Using Transcoding for Hidden Communication in IP Telephony*. Warsaw University of Technology, Institute of Telecommunications; 2011. Available from: <http://arxiv.org/pdf/1111.1250v1.pdf>
- [6] Stewart R. ed. *Stream Control Transmission Protocol*. – RFC 4960:6. Request for Comments: 4960, 2007. Available from: <http://tools.ietf.org/html/rfc4960>
- [7] Frączek W, Mazurczyk W, Szczypiorski K. *Stream Control Transmission Protocol Steganography*. Warsaw University of Technology, Institute of Telecommunications; 2010. Available from: <http://arxiv.org/abs/1006.0247>
- [8] ISO9646. *Open System Interconnection, Conformance Testing Methodology and Framework*. Switzerland, 1992

Multifactor Authentication Methods: A Framework for Their Comparison and Selection

Ignacio Velásquez, Angélica Caro and Alfonso Rodríguez

Abstract

There are multiple techniques for users to authenticate themselves in software applications, such as text passwords, smart cards, and biometrics. Two or more of these techniques can be combined to increase security, which is known as multifactor authentication. Systems commonly utilize authentication as part of their access control with the objective of protecting the information stored within them. However, the decision of what authentication technique to implement in a system is often taken by the software development team in charge of it. A poor decision during this step could lead to a fatal mistake in relation to security, creating the necessity for a method that systematizes this task. Thus, this book chapter presents a theoretical decision framework that tackles this issue by providing guidelines based on the evaluated application's characteristics and target context. These guidelines were defined through the application of an extensive action-research methodology in collaboration with experts from a multinational software development company.

Keywords: security, authentication scheme, multifactor authentication method, action-research, decision framework

1. Introduction

Generally, to protect the personal information of users in software applications, distinct authentication techniques are utilized to prevent intruders from accessing to it. Authentication is, thus, the process of verifying the identity of a user as part of a system's access control to protect the information stored within them [1]. Various authentication techniques have been proposed in literature, such as text passwords [2, 3], smart cards [4, 5], and biometrics [6–8]. All of the mentioned techniques belong to distinct authentication factors. An authentication factor is a piece of information that can be used to verify the identity of a user [9]. There are three main groups or factors of authentication techniques [10, 11]: (i) knowledge-based, that is, based on something that the user knows, such as text passwords; (ii) possession-based, that is, based on something that the user possesses, such as smart cards; and (iii) inherence-based, that is, something that the user is, such as biometrics. Two or more of these techniques can be combined to increase security, which is known as multifactor authentication [1].

In this book chapter, to differentiate between single-factor and multifactor authentication techniques, the former will be referred to as **authentication schemes**, whereas the latter will be referred to as **multifactor authentication methods**.

Nowadays, the decision of what authentication scheme or method to implement in a software application resides within the software development team. However, the experience of the involved developers can vary from team to team, which could affect in the decision of what authentication technique to implement. Due to the importance of security [12], selecting the wrong authentication technique could potentially be a fatal mistake [13].

The above statement creates the necessity of a method that systematizes the task of comparing and selecting the authentication schemes and methods. A few frameworks in literature partially help to achieve this [14, 15]; however, they do not present the adequate characteristics for their application in distinct application contexts or do not consider all authentication techniques or multifactor authentication. Thus, this book chapter presents a decision framework that covers the observed gap. This framework has been generated through the application of an action-research methodology [16]. This action-research has been performed in collaboration with a multinational software development company and contemplates the utilization of other research methodologies that support it.

The remainder of this book chapter is organized as follows. The methodology utilized for the research is presented in Section 2. Section 3 is focused on obtaining of the knowledge base utilized for the research. In Section 4, the generated decision framework is presented. Section 5 consists on the validation of the framework. Finally, the conclusions and future work of the research are given in Section 6.

2. Methodology

The realization of this research is within the scope of an action-research methodology that was carried for over a year in collaboration with a software development company. The objective of action-research is to provide a benefit for the research's "client" while also generating relevant "research knowledge" [16, 17]. This kind of collaboration allows to study complex social processes, such as the use of information technologies in organizations, by introducing changes in them and observing their effects [18].

There are four roles involved in action-research [19]. These roles are as follows:

- The **researcher(s)** who undertake(s) the action-research. In this case, the researchers are the book chapter's authors.
- The **studied object**, that is, the problem to solve. In this case, the studied object is the comparison and selection of authentication schemes and methods.
- The **critical group of reference** that has a problem that needs to be solved and also participates in the research process. In this case, the critical group of reference is composed by the employees of the partnered software development company (PSDC).
- The **beneficiary** who can receive benefits from the research results, without directly participating in its process. In this case, the main beneficiary is the PSDC, but other software developers can also benefit from this research.

During the realization of this action-research, multiple activities were performed in conjunction with the PSDC. These activities helped to generate and validate the proposed decision framework for solving the need of automatizing the comparison and selection of authentication techniques. These activities were performed utilizing the iterative process of action-research, which considers, for every cycle, the following four phases [20]: (i) the planning phase, which considers the elaboration of a research question to be answered through the iteration; (ii) the action phase, where distinct research methodologies are applied to address the posed research question; (iii) the observation phase, where the results of the interventions from the previous phase are processed; and (iv) the reflection phase, where the researchers share their findings with the group of reference to generate feedback; it is also possible to transversely perform this phase instead of cyclically [19], as it was done in this action-research through the realization of weekly progress meetings.

In this work, the action-research methodology was applied through three cycles. The objective of the first cycle was to obtain the required knowledge base for creating the framework. To achieve this, two strategies were applied: first, a systematic literature review (SLR) [21] was performed to obtain the existing knowledge in literature, and secondly, a number of surveys and interviews [16, 22] were conducted to learn the perceptions of the industry through the PSDC's employees. The second cycle was centered on the creation of the decision framework. During this cycle, an expert panel [23] was held to validate the initial draft of the framework. Finally, the third cycle focused on validating the final framework through the application of case studies [24].

3. Identification of the knowledge base

To construct the decision framework, it was necessary to obtain an adequate knowledge base regarding the topic at hand. To achieve this, two methodologies were applied. The first was the realization of a systematic literature review to identify the existing knowledge in related academic publications. The second corresponds to the application of a survey and interviews (S&I) to employees of the PSDC to learn the perceptions of the industry. The combined usage of these methods allowed the procurement of a knowledge base useful both for the academic and industrial sectors.

3.1 Systematic literature review

A systematic literature review has been carried out with the objective of "identifying authentication schemes proposed in literature and their possible combinations for their use as multifactor authentication methods, while also detecting criteria used for their comparison and selection and the existence of frameworks that handle such a task." Based on this objective, the following four research questions were formulated:

1. Which are the main authentication schemes that exist in the literature?
2. What combinations of these schemes can be found that can be used as multifactor authentication methods?
3. What criteria can be used to compare and/or to select between authentication schemes and/or multifactor authentication methods?

4. Are there frameworks that help to compare and/or to select authentication schemes or multifactor authentication methods? What are their characteristics?

The planning and results of the SLR have already been published in literature [25]. Additionally, a list containing the publications accepted during the SLR can be found in <http://colvin.chillan.ubiobio.cl/mcaro/>. Next, a brief summary of the main results of the SLR for every research question is presented.

3.1.1 Authentication schemes

A total of 515 publications regarding the proposal of authentication schemes were found. Their distribution among the authentication factors is as shown in **Figure 1**. Additionally, the context for which these schemes were proposed was recorded as well; this is presented in **Table 1**, including the publication's origin (journal article, conference article, or book chapter). It is important to mention that only 233 of the publications indicated a context.

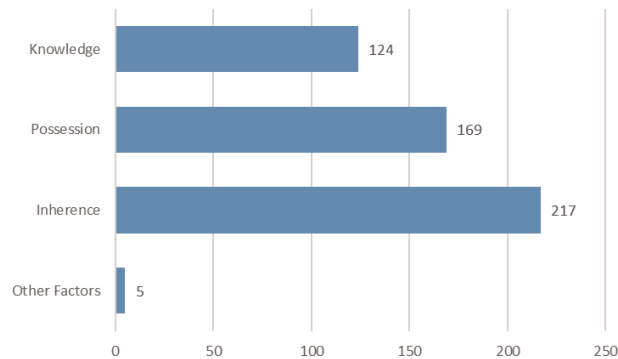


Figure 1.
Number of publications proposing authentication schemes for every authentication factor.

Context	Journal	Conference	Book	Total
Mobile environment	38	43	0	81
Remote authentication	31	11	0	42
Healthcare/telecare	23	1	0	24
Multi-server environment	15	2	0	17
Continuous authentication	9	2	0	11
Wireless sensor networks	8	2	0	10
Cloud computing	3	4	2	9
Banking and commerce	2	6	0	8
Smart environment	2	5	0	7
Login protocols	5	0	0	5
Web applications	4	1	0	5
Other contexts	7	7	0	14
Total	147	84	2	233

Table 1.
Number of publications proposing authentication schemes for every context.

3.1.2 Multifactor authentication methods

Four hundred forty-two publications proposing the combination of two or more authentication schemes in a multifactor manner were identified. Their distribution among the distinct authentication factor combinations is as shown in **Figure 2**. Similarly to the previous research question, the context for which these methods were proposed was recorded as well; this is presented in **Table 2**, including the publication's origin (journal article, conference article, or book chapter). In this case, 272 of the publications did indicate a context.

3.1.3 Comparison and selection criteria

Only 17 publications presented criteria for the comparison and selection of authentication schemes and methods. The presented criteria in the distinct publications can be categorized based on the kind of criteria proposed. Every publication

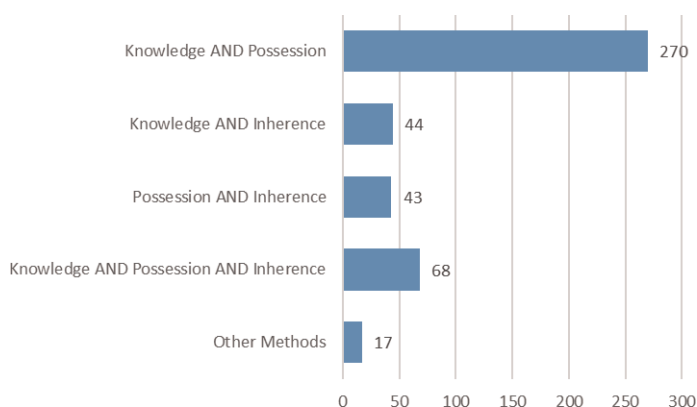


Figure 2.
 Publications proposing authentication methods for every factor combination.

Context	Journal	Conference	Book	Total
Remote authentication	52	12	0	64
Healthcare/telecare	45	3	0	48
Wireless sensor networks	29	4	0	33
Multi-server environment	22	7	0	29
Mobile environment	10	11	0	21
Cloud computing	12	5	0	17
Banking and commerce	6	5	0	11
Web applications	5	6	0	11
Wireless networks	6	2	0	8
USB devices	1	5	0	6
Insecure environment	3	2	0	5
Other contexts	15	3	1	19
Total	206	65	1	272

Table 2.
 Number of publications proposing authentication methods for every context.

considered one or more criteria categories; however, only three of them could be identified in more than one publication. The most identified categories of criteria are usability, security, and costs. The first two were identified in nine publications each, whereas the latter was found in five publications.

Moreover, it could be observed that most of these articles highly considered the importance of the use context for comparing and selecting schemes and methods. This was mainly done by the publication addressing specific contexts or considering the context itself as another criterion.

3.1.4 Decision frameworks

Eight decision frameworks that help in the comparison and selection of authentication schemes and methods were identified. Through the analysis of these frameworks, it could be observed that multifactor authentication is not often considered, whereas proposals that do consider it utilize a limited number of criteria. Thus, no decision framework that considered multifactor authentication and enough criteria for a detailed comparison and selection of authentication schemes and methods could be found.

3.2 Survey and interviews

A survey and interviews have been applied to the PSDC's employees with the objective of learning the perceptions of people from the industry regarding authentication and the comparison and selection of distinct schemes and methods. The interviews were realized as a pilot application of the survey. A total of 12 employees were interviewed. In addition, 45 valid responses, out of a sample of 83 people ranging from developers to project leads, were received through the survey. Out of the 57 respondents, over two thirds of them held a senior position in the PSDC, as well as having over 6 years of working experience.

Four main questions were posed to the respondents, whose contents can be summarized as follows:

Q1. What authentication schemes do you know?

Q2. What multifactor authentication methods do you know?

Q3. What authentication schemes or multifactor authentication methods have you implemented in applications that you have developed?

Q4. What is the importance that you give to distinct factors when deciding what authentication scheme or method should be implemented in an application?

In <http://colvin.chillan.ubiobio.cl/mcaro/> it is possible to find the questionnaire used for the survey. A summary of the responses obtained for every question is provided next.

3.2.1 Authentication schemes known by the respondents

For this question, respondents were asked to mark from a list the authentication schemes that they knew. The most known schemes were text passwords, one-time passwords (OTP, tokens), and mobile-based authentication. All respondents answered this question. The complete results of this question can be observed in **Table 3**, which shows the number of survey respondents and interviewed people that know each authentication scheme.

Authentication scheme	Interviewees	Survey respondents
Text passwords (TP)	10	40
Graphical passwords (GP)	1	20
Cognitive authentication (CA)	0	10
OTP (tokens)	7	38
Smart cards (SC)	3	24
Mobile-based (MB)	8	31
Biometrics (B)	5	30
Federated single sign-on (FSSO)	4	22
Proxy-based (PB)	1	8
Others	0	2

Table 3.
 Number of respondents that know each authentication scheme.

3.2.2 Multifactor authentication methods known by the respondents

For the second question, respondents were given a brief explanation about multifactor authentication. Afterward, they were asked what multifactor authentication methods they knew. The combination of text passwords and OTP was the most known among them. A total of 27 out of the 45 survey respondents answered this question. The complete results of this question can be observed in **Table 4**, which shows the number of survey respondents and interviewed people that know each multifactor authentication method.

Combination	Method	Interviewees	Survey respondents
Knowledge + possession	TP + OTP	7	15
	TP + SC	2	8
	TP + MB	6	6
	Others	0	1
	Total	15	30
Knowledge + inheritance	TP + B	0	15
	Others	0	3
	Total	0	18
Possession + inheritance	OTP + B	0	6
	MB + B	0	3
	SC + B	0	3
	Total	0	12
Knowledge + possession + inheritance	TP + SC + B	0	7
	TP + OTP + B	1	2
	Others	0	2
	Total	1	11
Grand total		16	71

Table 4.
 Number of respondents that know each authentication method.

3.2.3 Authentication schemes and methods implemented by the respondents

Next, the respondents were asked what authentication techniques they had implemented in applications developed by them and the kind of application. Most applications were either web-based or for banking and commerce. A total of 23 out of the 45 survey respondents answered this question. The complete results of this question can be observed in the graphs of **Figures 3 and 4**, which show the implemented authentication schemes and methods and the contexts of the applications that were being developed, respectively.

3.2.4 Comparison and selection criteria used by the respondents

For the last question of the S&I, distinct strategies were applied between the interviewees and the survey respondents. In the case of the former, they were directly asked what criteria they utilized for the comparison and selection of authentication schemes and methods. In the case of the latter, the responses from the interviewees, coupled with the results of the previously performed SLR, were used to generate a list of comparison and selection criteria that respondents were asked to value from 1 to 5. A higher value meant that the respondent gave a higher importance to the criterion. A total of 29 out of the 45 survey respondents answered this question. The complete results of this question can be observed in **Table 5** and

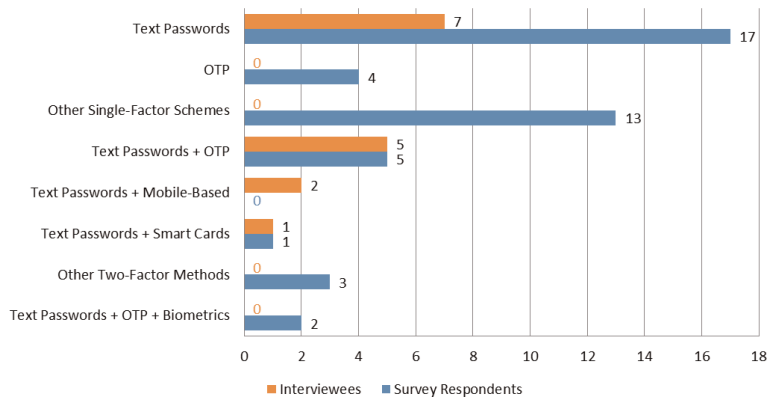


Figure 3.
Authentication schemes and methods implemented by the respondents.

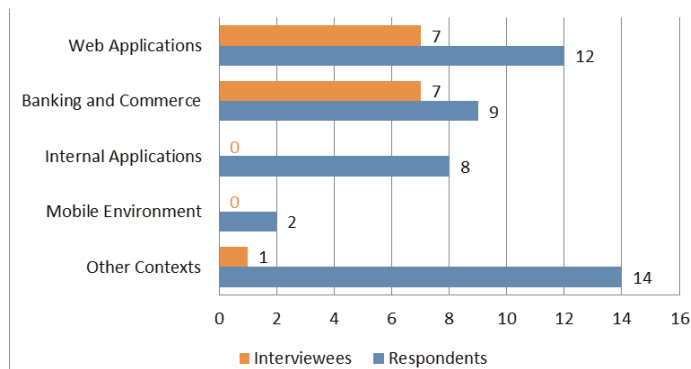


Figure 4.
Contexts of the applications developed by the respondents.

Criterion	Interviewees that consider the criterion
Client's requirements	11
Application context	11
Usability-related criteria	9
Security-related criteria	11
Cost-related criteria	8
Other criteria	2

Table 5.
 Comparison and selection criteria considered by the interviewees.

in **Table 6**, which show the responses given by the interviewees and the survey respondents, respectively.

Finally, survey respondents were asked what other comparison and selection criteria they would consider. The received answers include the ease of authentication information recovery, the registration method, and the sensitivity of the information.

3.3 Short survey

A second survey was later applied to nine employees of the PSDC. These employees were selected among the most experienced developers of the company, based on their years of experience and positions. The single aim of this survey was to ascertain the importance that the respondents would assign to an application's security and usability based on the target context. The importance was valued in percentages, with the sum of usability and security being 100% for every context. **Table 7** presents the results of this survey.

The obtained values were used afterward as part of the input for the decision framework.

Category	Criterion	Value
Usability	Ease of use	3.31
	Ease of learning	3.28
	Need of using a device	3.10
	Method's reliability	4.10
Security	Importance of security	4.41
	Resistance to well-known attacks	4.21
Costs	Implementation costs	4.07
	Costs per user	4.00
	Server compatibility	3.69
	Need of acquiring licenses	3.86
	Available technologies	3.93
Others	Client's requirements	4.17
	Application context	4.41
	Norms and legislation	3.90

Table 6.
 Comparison and selection criteria valued by the survey respondents.

Context	Importance of security (%)	Importance of usability (%)
Mobile environment	45.56	54.44
Remote authentication, multi-server environment, cloud computing	64.44	35.56
Healthcare/telecare	57.78	42.22
Wireless sensor networks	63.33	36.67
Banking and commerce	73.33	26.67
Web applications	28.89	71.11

Table 7.
Importance given to security and usability in distinct contexts by the respondents.

4. The framework

This section describes the decision framework constructed through the knowledge base acquired by using the methodologies presented above. It has been given the name of Kontun framework, which means “to enter foreign property” in Mapudungún, an indigenous language from Chile, which is what it aims to prevent. **Table 8** shows a summary of the main findings during the knowledge base gathering and their origin (either the SLR or the S&I).

A summary of the constructed framework’s characteristics is provided next. A complete description can be found in [26].

First, the framework considers a number of criteria obtained from the knowledge base, divided among the three most observed categories: security, usability, and costs. Each criterion is then given distinct possible importance values and a weight based on the findings from the knowledge base. To illustrate the above

Most reported knowledge-based schemes	<ul style="list-style-type: none"> • Text passwords (SLR, S&I) • Graphical passwords (SLR)
Most reported possession-based schemes	<ul style="list-style-type: none"> • Smart cards (SLR) • OTP (S&I) • Mobile-based (S&I)
Most reported inheritance-based schemes	<ul style="list-style-type: none"> • Face biometrics (SLR, S&I) • Behavioral biometrics (SLR) • Palm print (SLR) • Fingerprints (SLR, S&I) • Vein biometrics (SLR) • Iris biometrics (SLR, S&I)
Multifactor authentication	<ul style="list-style-type: none"> • Prevalence of the combination of knowledge- and possession-based authentication schemes (SLR, S&I)
Most observed application contexts	<ul style="list-style-type: none"> • Mobile environment (SLR) • Remote authentication (SLR) • Multi-server environment (SLR) • Cloud computing (SLR) • Healthcare/telecare (SLR) • Wireless sensor networks (SLR) • Banking and commerce (S&I) • Web applications (S&I)
Comparison and selection criteria	<ul style="list-style-type: none"> • Criteria are mainly related to usability, security, and costs (SLR) • Identified criteria are valued positively by the industry (S&I) • High importance observed regarding application context (SLR, S&I)

Table 8.
Summary of the acquired knowledge base.

criterion, **Table 9** shows the usability-related criteria, their importance values, and their weights.

Every criterion has two or more importance values between 20 and 100, and the sum of all the weights of the criteria belonging to the same category is 100%. In this manner, when using the framework, a person must select the importance values that best describe their application and then calculate the average values of security (S), usability (U), and costs (C) using the following equations:

$$S = \sum_{\text{for each criterion of } S} \text{AssessmentValue} * \text{CriterionWeight} \quad (1)$$

$$U = \sum_{\text{for each criterion of } U} \text{AssessmentValue} * \text{CriterionWeight} \quad (2)$$

$$C = \sum_{\text{for each criterion of } C} \text{AssessmentValue} * \text{CriterionWeight} \quad (3)$$

The framework also considers a number of common contexts identified through the knowledge base. These contexts were given distinct weights based on the importance of security and usability in the context itself. Here, a term known as the security/usability value (SUV) is presented. The knowledge base allowed to ascertain the fact that, generally, the more secure an authentication scheme or method is, it has a lower usability and vice-versa. The SUV is used to denotate this. Based on the calculated average values of S, U, and C, coupled with the selected application context (Ct), the SUV is calculated as follows:

$$SUV = A * S + B * (100 - U) \quad (4)$$

A and B are constants defined based on the importance given to S and U, respectively, in the selected context. A high SUV value thus indicates that more

Criterion	Importance	Value	Weight
Ease of use	The method necessarily needs to be easy to use	100	25%
	The method preferably needs to be easy to use	60	
	It is not necessary for the method to be easy to use	20	
Ease of learning	A user should not take longer than a day to get used	100	25%
	A user should not take longer than a week to get used	60	
	The time it takes to get used is not relevant	20	
Authentication information recovery	The recovery process should be simple	100	10%
	The recovery process should be complex	20	
Need of using a device	It does not need to use a device	100	10%
	It can use a possession or biometric device	60	
	It can use both a possession and a biometric device	20	
Authentication method's reliability	It should never or hardly fail during authentication	100	30%
	It should not fail occasionally during authentication	75	
	It can fail occasionally during authentication	45	
	It does not matter how often it fails	20	

Table 9.
Criteria considered by the framework.

secure authentication methods should be implemented in the application, whereas a low SUV indicates that more usable authentication schemes or methods should be implemented in the application.

Having calculated the SUV and also considering the average value given to C, the framework is able to provide a suggestion on what authentication schemes or methods to implement in the evaluated application. The recommendation is as follows: for a SUV of 65 or higher, the framework will suggest the implementation of highly secure authentication methods; for a SUV of 35 or lower, the framework will suggest the implementation of highly usable authentication schemes; and for a SUV between 35 and 65, the framework will suggest the implementation of averagely secure and usable authentication methods. Moreover, for a value of C of 60 and above, the framework will suggest the implementation of more affordable authentication schemes or methods; for a value of C below 60, the framework will suggest the implementation of more expensive authentication schemes or methods. The recommendations are also different based on the target Ct. Thus, for every Ct, the framework will give six possible recommendations based on the calculated SUV and C. **Table 10** illustrates the above framework for the context of mobile environment.

Finally, the person utilizing the framework must decide the authentication scheme or method to implement in their application, taking into consideration the recommendations given by the framework.

4.1 Tool prototype

To facilitate the use of the framework in software development environments, a tool prototype has been constructed that allows its utilization in a semiautomatic manner. This tool has also supported the validation process of the framework. With the tool prototype, the person in charge only needs to indicate the evaluated application's features and target context through a radio form. Afterward, the tool prototype automatically calculates the values of average S, U, and C and the SUV. The tool prototype is available for download in <http://colvin.chillan.ubiobio.cl/mcaro/>.

SUV 65 C < 60	Graphical passwords + smart cards + behavioral biometrics Text passwords + OTP + behavioral biometrics Graphical passwords + OTP + behavioral biometrics Graphical passwords + OTP + face biometrics
SUV 65 C 60	Text passwords + smart cards + behavioral biometrics Text passwords + smart cards + face biometrics
35 < SUV < 65 C < 60	Graphical passwords + behavioral biometrics OTP + behavioral biometrics Text passwords + palm print/fingerprints Graphical passwords + OTP
35 < SUV < 65 C 60	Text passwords + behavioral biometrics Text passwords + smart cards
SUV 35 C < 60	Behavioral biometrics Graphical passwords Face biometrics Palm print/fingerprints
SUV 35 C 60	Behavioral biometrics Text passwords Graphical passwords

Table 10.
Recommendation given by the framework for the context of mobile environment.

The tool prototype has been developed using the model view controller (MVC) design pattern, with the Java programming language and supported by the Spring Framework. PostgreSQL has been used as the database management system.

The main screens of the tool prototype can be observed in **Figures 5–7**. They show the procedures for the criteria selection, the context selection, and the framework's recommendation, respectively.

DEFINE CRITERIA

Usability - Security - Costs

Ease of Use ?	<input type="radio"/> The method necessarily needs to be easy to use. <input type="radio"/> The method preferably needs to be easy to use. <input type="radio"/> It is not necessary for the method to be easy to use.
Ease of Learning ?	<input type="radio"/> A user should take no longer than a day to get used to using the method. <input type="radio"/> A user should take no longer than a week to get used to using the method. <input type="radio"/> The time that a user takes to get used to using the method is not relevant.
Authentication Information Recovery ?	<input type="radio"/> The authentication information recovery process should be simple. <input type="radio"/> The authentication information recovery process should be complex.
Need of Using a Device ?	<input type="radio"/> The method does not need the use of a device. <input type="radio"/> The method can need the use of either a possession device or a biometric device. <input type="radio"/> The method can need the use of both a possession device and a biometric device.
Authentication Method's Reliability ?	<input type="radio"/> The method should never or hardly fail during authentication. <input type="radio"/> The method should not fail occasionally during authentication. <input type="radio"/> The method can fail occasionally during authentication. <input type="radio"/> It does not matter how often the method fails during authentication.

Figure 5.
Criteria selection in the tool prototype.

SELECT CONTEXT

Context	Usability Weight	Security Weight
<input type="radio"/> Mobile Environment ?	Medium	Medium
<input type="radio"/> Remote Authentication, Multi Server Environment and Cloud Computing ?	Low	High
<input type="radio"/> Healthcare / Telecare ?	Medium	Medium
<input type="radio"/> Wireless Sensor Networks ?	Low	High
<input checked="" type="radio"/> Banking and Commerce ?	Low	High
<input type="radio"/> Common Web Applications ?	High	Low
<input type="radio"/> Other Context	Medium	Medium

Figure 6.
Context selection in the tool prototype.

RECOMMENDATION

Given the previously selected criteria and in order from the most recommended one to the least, it is recommended that you implement one of the following authentication methods in your application:

- Text Passwords and One Time Passwords
- Mobile Based Authentication and Behavioral Biometrics
- One Time Passwords and Behavioral Biometrics

The above considering a **medium Usability**, a **low Security**, **high Costs** and the context of **Banking and Commerce**.

For a brief description of every authentication method mentioned above, you can go [here](#)

Figure 7.
Framework's recommendation in the tool prototype.

The tool prototype also has additional features that facilitate its use in software development companies. Specifically, it has a user registration feature which allows maintaining a registry of its usage and a functionality for adapting its preferences based on the software development company's needs.

5. Validation through the industry

Through the creation of the framework, its adequacy was repeatedly validated using strategies associated to the application of the action-research methodology. Specifically, the validation was ascertained through the realization of an expert panel and the application of case studies. These are detailed in remainder of this section.

5.1 Expert panel

An expert panel was held in collaboration with five experts from the PSDC that consisted of four sessions with the aim of ascertaining their perceptions regarding an initial draft of the framework, so that it was more adequate to the real requirements observed in a software development environment. The activities during every session of the expert panel are described next.

5.1.1 *Presentation of the initial draft of the framework*

The first session consisted on the presentation of the initial draft of the framework, with the purpose of helping the experts to have a general notion of the aim of this research.

5.1.2 *Validation of comparison and selection criteria*

The preliminary list of criteria, their categorization, their values, and their weights were presented to the experts for their validation. This allowed to discard the least adequate ones and to generalize those that were too specific for the needs of a software development team.

5.1.3 *Validation of the considered contexts*

The contexts considered by the framework were presented to the experts. Similarly to the previous session, this allowed to make the appropriate modifications to the currently selected contexts. Additionally, the SUV was presented to the experts, who generally agreed to the adequacy of its use.

5.1.4 *Validation of the framework's recommendations*

The authentication schemes and methods recommended for every situation were presented to the experts. This allowed to ascertain the adequacy of every recommendation. The experts were generally in agreement with the recommendations.

5.2 Case studies

After its construction, the validation of the framework's recommendations was realized through the application of a case study methodology in collaboration with

the PSDC. Specifically, the framework's recommendations were compared with the authentication schemes or methods implemented in existing applications developed by the PSDC or with the recommendations that their experts would give for hypothetical situations. The case studies are described in detail in [26]. Next, a brief summary of their application is provided.

The case studies are split in three categories: (i) those that were realized by comparing the framework's recommendation against the implemented scheme or method on an existing application, (ii) those that were realized by comparing the framework's recommendation against the recommendations given by experts for hypothetical applications, and (iii) those that were realized by comparing the framework's recommendation against the implemented scheme or method on an existing application and also against the recommendation given by experts for hypothetical applications with nearly the same features as the existing ones. These case studies are presented in **Tables 11–13**, respectively, presenting the implemented scheme or method in the existing application, the framework's recommendation, the most recommended scheme or method by the experts, and the acceptance rate of the framework's recommendation, as appropriate.

In general, the results of the case studies are favorable for the framework. It is important to mention that, where discrepancies are observed, there was often a reasoning behind them. For example, for case study 3 (existing application), the implemented scheme was demanded by the client and not selected by the software development team.

ID	Implemented scheme or method	Framework's recommendation
1	Two-factor authentication (text passwords + smart cards)	Three-factor authentication (text passwords + OTP + behavioral biometrics)
2	Two-factor authentication (text passwords + mobile-based)	Two-factor authentication (text passwords + mobile-based)
3	OTP (demanded by client)	Behavioral biometrics

Table 11.
Case studies based on existing applications.

ID	Experts' recommendation	Framework's recommendation	Acceptance rate of framework's recommendation
4	Two- or three-factor authentication	Three-factor authentication	100%
5	Text passwords	Two-factor authentication	80%

Table 12.
Case studies based on hypothetical applications.

ID	Implemented scheme or method	Experts' recommendation	Framework's recommendation	Acceptance rate of framework's recommendation
6	Two-factor authentication	Text passwords	Text passwords	100%
7	Text passwords	Two-factor authentication	Two-factor authentication	90%

Table 13.
Case studies based on existing applications with a hypothetical counterpart.

6. Conclusions

The research presented in this book chapter summarizes the definition of a theoretical framework. This framework will help in the comparison and selection of the most appropriate authentication schemes or multifactor authentication methods for applications created by software developers. It has been created through the application of an action-research methodology that considered the utilization of various other research methodologies that helped to contribute in distinct ways to the research objective.

On the one hand, a systematic literature review, coupled with surveys and interviews, was performed to obtain the required knowledge base for generating the framework. The utilization of these two methodologies allowed to ascertain the perceptions on authentication from both the academy and the industry.

On the other hand, an expert panel and several case studies were realized to validate the adequacy of the framework. This permitted to obtain feedback from the end users of the framework so that it would provide adequate authentication scheme or method recommendations and have an appropriate usability.

Thus, this experience allowed to observe the usefulness of performing a research in collaboration with the industry, as it permits obtaining results that align more adequately with their needs while also providing more refined academic results.

Several future work lines can be followed based on this research. Namely, the framework could be adapted to work as a recommendation system so that its recommendations get refined through its usage. For the industry, it would be of interest that the framework not only recommends an authentication technique but that it also provides the required code for its implementation. Finally, the last cycle of the action-research, that is, the realization of case studies, could be replicated in other software development companies to further validate the adequacy of the framework.

Acknowledgements

This research is part of the following projects: DIUBB 144319 2/R and BuPERG (DIUBB 152419 G/EF).

Author details

Ignacio Velásquez, Angélica Caro* and Alfonso Rodríguez
Computer Science and Information Technologies Department, University of
Bío-Bío, Chillán, Chile

*Address all correspondence to: mcaro@ubiobio.cl

IntechOpen

© 2019 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] O’Gorman L. Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*. 2003;**91**(12):2021-2040
- [2] Kumari S, Khan MK, Li X, Wu F. Design of a user anonymous password authentication scheme without smart card. *International Journal of Communication Systems*. 2016;**29**(3): 441-458
- [3] Ranjan P, Om H. An efficient remote user password authentication scheme based on Rabin’s cryptosystem. *Wireless Personal Communications*. 2016:1-28
- [4] Yang TC, Lo NW, Liaw HT, Wu WC. A secure smart card authentication and authorization framework using in multimedia cloud. *Multimedia Tools and Applications*. 2017;**76**(9):11715-11737
- [5] Mishra D, Das AK, Mukhopadhyay S. A secure and efficient ECC-based user anonymity-preserving session initiation authentication protocol using smart card. *Peer-to-Peer Networking and Applications*. 2016;**9**(1):171-192
- [6] Samangouei P, Patel VM, Chellappa R. Facial attributes for active authentication on mobile devices. *Image and Vision Computing*. 2017;**58**:181-192
- [7] Antal M, Szabó LZ. Biometric authentication based on touchscreen swipe patterns. *Procedia Technology*. 2016;**22**:862-869
- [8] Usha K, Ezhilarasan M. Robust personal authentication using finger knuckle geometric and texture features. *Ain Shams Engineering Journal*. 2016;**9**(4):549-565
- [9] Jacomme C, Kremer S, editors. An extensive formal analysis of multi-factor authentication protocols. In: 2018 IEEE 31st Computer Security Foundations Symposium (CSF); IEEE. 2018
- [10] Colnago J, Devlin S, Oates M, Swoopes C, Bauer L, Cranor L, et al., editors. “It’s not actually that horrible”: Exploring adoption of two-factor authentication at a university. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*; ACM. 2018
- [11] Huang X, Xiang Y, Chonka A, Zhou J, Deng RH. A generic framework for three-factor authentication: Preserving security and privacy in distributed systems. *IEEE Transactions on Parallel and Distributed Systems*. 2011;**22**(8):1390-1397
- [12] Easttom II WC. *Computer Security Fundamentals: Pearson IT Certification*; 2019
- [13] Nissanke N, Khayat EJ, editors. *Risk Based Security Analysis of Permissions in RBAC*. WOSIS; 2004
- [14] Bonneau J, Herley C, Van Oorschot PC, Stajano F, editors. The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In: 2012 IEEE Symposium on Security and Privacy; IEEE. 2012
- [15] Forget A, Chiasson S, Biddle R. User-centred authentication feature framework. *Information and Computer Security*. 2015;**23**(5):497-515
- [16] Genero M, Cruz-Lemus J, Piattini M. *Métodos de investigación en ingeniería del software*. Madrid, Spain: Editorial RA-MA; 2014. pp. 171-199
- [17] Kock N, Lau F. Information systems action research: Serving two demanding masters. *Information Technology & People*. 2001;**14**(1)
- [18] Eden C, Ackermann F. Theory into practice, practice to theory: Action research in method development.

European Journal of Operational Research. 2018;**271**(3):1145-1155

[19] Wadsworth Y. What Is Participatory Action Research? Action Research Issues Association; 1993

[20] Padak N, Padak G. Guidelines for planning action research projects. Research to Practice. ERIC. 1994

[21] Kitchenham B. Procedures for performing systematic reviews. Keele, UK, Keele University. 2004;**33**(2004): 1-26

[22] Kitchenham BA, Pfleeger SL. Personal opinion surveys. In: Guide to Advanced Empirical Software Engineering. Springer; 2008. pp. 63-92

[23] Rosqvist T, Koskela M, Harju H. Software quality evaluation based on expert judgement. Software Quality Journal. 2003;**11**(1):39-55

[24] Runeson P, Host M, Rainer A, Regnell B. Case Study Research in Software Engineering: Guidelines and Examples. John Wiley & Sons; 2012

[25] Velásquez I, Caro A, Rodríguez A. Authentication schemes and methods: A systematic literature review. Information and Software Technology. 2018;**94**:30-37

[26] Velásquez I, Caro A, Rodríguez A. Kontun: A framework for recommendation of authentication schemes and methods. Information and Software Technology. 2018;**96**:27-37