Available online at www.sciencedirect.com

SciVerse ScienceDirect

journal homepage: www.elsevier.com/locate/ijcip

Analysis of the security of VPN configurations in industrial control environments

*Sanaz Rahimi**, *Mehdi Zargham*

Department of Computer Science, Southern Illinois University, 1000 Faner Drive, Carbondale, Illinois 62901, USA

ARTICLE INFO

Article history:
Received 19 May 2011
Published online 28 January 2012

Keywords:
Control systems
Virtual private networks
Security analysis
Simulation

ABSTRACT

Virtual private networks (VPNs) are a popular approach for protecting otherwise insecure industrial control protocols. VPNs provide confidentiality, integrity and availability, and are often considered to be secure. However, implementation vulnerabilities and protocol flaws expose VPN weaknesses in many industrial deployments. This paper employs a probabilistic model to evaluate and quantify the security of VPN configurations. Simulations of the VPN model are conducted to investigate the trade-offs and parameter dependence in various VPN configurations. The experimental results provide recommendations for securing VPN deployments in industrial control environments.

1. Introduction

Virtual private networks (VPNs) are widely used to ensure secure communications over insecure public networks. VPNs provide security services such as confidentiality, integrity and availability by creating encrypted tunnels between the communicating parties.

VPNs are recommended in the literature and by many critical infrastructure protection standards to secure process control, supervisory control and data acquisition (SCADA) and automation protocol communications [1–6]. Although these protocols are generally very reliable, they were not designed to resist malicious attacks. As a result, it is recommended to wrap industrial protocols such as DNP3 [7], 61850 [8] and Modbus [9] within VPN tunnels to protect them from unauthorized access. These configurations supposedly offer confidentiality, integrity and availability [6], but little work has focused on the secure configuration of VPN tunnels and the maintenance required for their secure operation [6].

VPNs are attractive targets for attackers. Because VPNs carry sensitive information over public networks, successfully breaking into a VPN tunnel enables an attacker to alter sensitive data and commands without physical access to the industrial facility. If other protection mechanisms such as strong access control are not deployed properly, the attacker can gain access to the internal SCADA systems through the VPN tunnel. Also, as industrial control systems implement more security mechanisms, VPNs can become the weakest link in the security chain.

VPNs have several vulnerabilities. According to Hills [10], most VPN implementations suffer from serious security flaws that can be easily exploited by attackers to fabricate, intercept, modify and interrupt traffic. Some of these vulnerabilities are implementation specific; they are the result of flaws in a specific protocol implementation due to bad coding, incomplete implementation or poor implementation choices for conditions that are not specified in the standard. However, vulnerabilities in the underlying protocols cannot be addressed by good implementation [10]. As recent incidents have shown, sophisticated malware [11] can stealthily modify the configurations of a control system (including a VPN) and seriously impact its operation. The solutions to these security problems are proper configuration, continual configuration validation and regular maintenance, all of which are

* *Corresponding author.*
E-mail addresses: srahimi@siu.edu (S. Rahimi), mehdi@siu.edu (M. Zargham).

1874-5482/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.ijcip.2012.01.001

effective only if system administrators fully understand and analyze the internal details of the protocols.

This paper models VPNs using stochastic activity networks [12] (an extended form of Petri nets [13]) and analyzes the probability of successful breaches against various VPN configurations. VPN security is quantified for different choices of parameters such as key length, mode of operation, number of users and maintenance frequency. The results provide recommendations for securely deploying VPNs in industrial control environments.

2. VPN vulnerabilities

VPNs are categorized according to their layer, i.e., transport layer (SSL), network layer (IPSec) and link layer (L2TP). This paper focuses on IPSec VPNs. As an underlying VPN protocol, IPSec [14] provides confidentiality and integrity services in the network layer (i.e., on a per packet basis) using two main subprotocols (AH and ESP) in two different modes (transport and tunnel). The detailed description of IPSec is beyond the scope of this paper. We only describe the features that are relevant to our discussion. Interested readers are referred to [14] for details about IPSec.

2.1. IPSec

IPSec provides security services via the Authentication Header (AH) and Encapsulation Security Payload (ESP) protocols. AH provides integrity by adding the hash-based message authentication code (HMAC) [15] of the entire packet (payload and full IP header); however, it does not provide confidentiality because it leaves the packet in plaintext. ESP encrypts the packet payload and certain fields of the IP header, and adds the ESP header and trailer to the IP packet, providing confidentiality and limited integrity.

IPSec can operate in the transport or tunnel modes. The transport mode is used when end-to-end security is desired and both end nodes support IPSec. In the transport mode, the original IP header is preserved for routing purposes and the AH/ESP headers are added under the IP header. The tunnel mode is used when the end machines do not support IPSec or when the identities of the communicating entities have to be hidden. In the tunnel mode, the entire IP packet is encrypted and a new IP header is added to the packet. The gateway on the border of each organization provides the security services by adding and removing IPSec headers.

Security Association (SA) is the concept used in IPSec for connection management between two communicating entities. An SA comprises a secure communication channel and its parameters, including the encryption algorithms, keys and lifetimes. Each SA is unidirectional and can provide one security service (AH or ESP). Two SAs are required for bidirectional communications.

IPSec uses the Internet Key Exchange (IKE) protocol to manage and exchange encryption keys and algorithms. IKE is a hybrid of three sub-protocols: Internet Security Association and Key Management Protocol (ISAKMP), Versatile Secure Key Exchange Mechanism for Internet (SKEME) and Oakley. ISAKMP provides the framework for authentication and SA management, but does not define the specific algorithms and keys. IKE uses the SKEME and Oakley protocols for key exchange and agreement with the acceptable cryptographic algorithms.

Because IKE is commonly used to establish VPN channels, many VPN vulnerabilities are in one way or another related to it. We provide an overview of IKE and its modes of operation to provide a better understanding of VPN vulnerabilities.

IKE has three important modes: main mode, aggressive mode and quick mode. The main mode is used for authentication and key exchange in the initial phase of IKE. This phase assumes that no SA is present and that the two parties wish to establish SAs for the first time. It involves three pairs of messages. The first pair negotiates the security policy and encryption algorithms to be used. The second pair establishes the keys using the Diffie–Hellman key exchange protocol. The third pair authenticates peers using signatures or certificates. Note that the identities of the peers in the main mode are often their IP addresses.

The aggressive mode is also used for the initial key exchange, but it is faster and more compact than the main mode. This mode involves a total of three messages that contain the main mode parameters, but in a more compact form. Key and policy exchange are performed by the first two messages while the third message authenticates the initiator to the responder. Note that the identity of the responder (sent in the second message) is not protected, which is a vulnerability in the aggressive mode of operation.

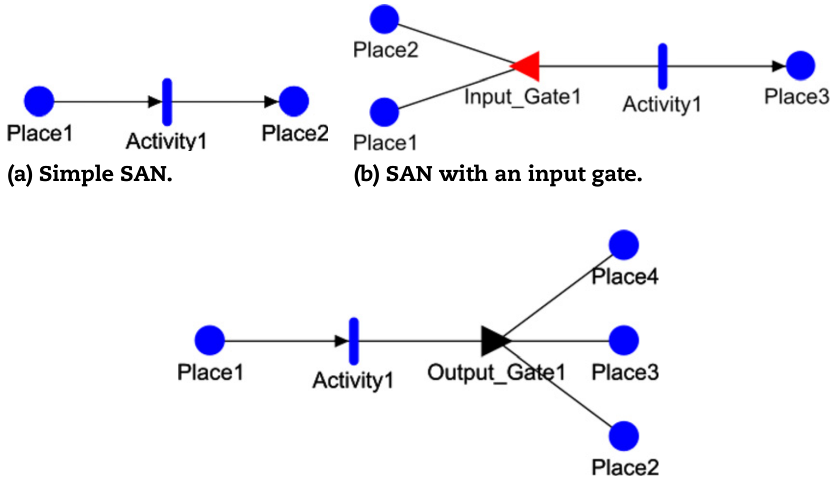
The quick mode is used for the negotiations in the secondary phase of IKE. This mode assumes that the peers have already established SAs and that the exchange can update the parameters or renew the keys. The quick mode messages are analogous to those in the aggressive mode, but the payloads are encrypted. If the quick mode operates with the perfect-forward-secrecy option, the shared secrets are renewed with a fresh Diffie–Hellman exchange.

IKE authenticates peers using a pre-shared key (PSK), public key encryption or digital signature. In the PSK method, which corresponds to traditional username/password authentication, the peers share a secret through a back channel and exchange the hash value of the secret for authentication. Unfortunately, although this method has known vulnerabilities, it is the only mandatory authentication method according to the RFC [16]. Public key encryption is another method of authentication in which the peers digitally sign their identities; however, the keys are required to be provided in advance by some other means. Digital certificates may also be used for authentication in IKE; in this mode, the peers exchange certificates to mutually authenticate each other.

2.2. VPN vulnerabilities

VPNs have several vulnerabilities. The common username enumeration vulnerability refers to an implementation flaw in which the username/password authentication mechanism responds differently to invalid usernames and passwords. By exploiting this vulnerability, an attacker can identify valid usernames that can be used later in password discovery.

When IKE is used in the aggressive mode with a pre-shared key (PSK), the client sends an IKE packet to the server



(a) Simple SAN.

(b) SAN with an input gate.

(c) SAN with an output gate.

Fig. 1 – Simple SAN abstractions.

containing, among other things, the identity (username) of the client. The server then responds with another packet that contains the hash of the identity and the PSK (password). When an incorrect username is received, many VPN implementations send an error message, send a packet with NULL PSK or do not respond at all. This enables an attacker to infer whether or not a username exists by sending a single packet (enumerate username). Upon discovering the first username, the attacker can generate likely usernames with the same pattern (e.g., first letter of the first name concatenated with the last name). When a VPN is used in the main mode, the identity is an IP address, not a username.

Hills [10] proposes that a secure VPN implementation returns the hash of a randomly-generated password whenever it receives an invalid username. However, this does not solve the problem because an attacker can still send two different packets with the same username; if two different hashes are received, then the attacker knows that the username does not exist and vice versa. Furthermore, the attacker can delay the two packets using a number of packets for other usernames to flush a buffer that the server may employ to track such an attack. The solution is for the server to encrypt the username with a secret key (generated and kept on the server only for this purpose) and to return the hash of this value. Thus, the server always responds to a username with a unique hash value, which foils the attack.

When the attacker discovers a valid username, he/she can receive the hash of the password associated with the username from the server (using PSK in the aggressive mode). The attacker can then apply an offline algorithm to crack the hash value and obtain the password. Offline cracking can be very fast because the probabilistic model of VPN password hashing is not hidden. This poses a serious threat to short passwords. A vulnerability exists even if IKE operates in the main mode with PSK; it can be exploited by a man-in-the-middle attack (e.g., using DNS spoofing [17]) to gain the Diffie–Hellman shared secrets. The only difference is that, in the main mode, the identity of each peer is its IP address.

When a username/password pair is successfully found, the first phase of IKE is breached. If the VPN configuration does not require extra authentication, the breach is enough to setup a VPN channel with the server. In some other cases, the configuration requires an extra XAUTH step to complete the second phase of IKE, but this phase is vulnerable to a man-in-the-middle attack as mentioned in the standard [18]. The reason for the vulnerability is that XAUTH must be used in conjunction with the first phase of IKE; if this phase is not performed properly, XAUTH cannot provide any security guarantees. Therefore, an attacker who performs a man-in-the-middle attack would be able to authenticate successfully.

Other VPN implementation vulnerabilities include VPN fingerprinting (inferring information about a device from its behavior); insecure password storage (e.g., in the registry or as plaintext in memory); lack of account lockout; poor default configurations; and unauthorized modification of configurations. We do not consider these vulnerabilities in this paper because they are implementation specific and may require other flaws to be exploited in a successful attack. For example, insecure password storage could be exploited by the installation of malware.

3. Modeling methodology

Our approach is to use a modeling abstraction to study the probabilistic cracking of a VPN connection. The model that is created is solved quantitatively using simulation.

We model the security of VPNs using an extension of Petri nets [13] called Stochastic Activity Networks (SANs) [12]. The Mobïus [19] tool is used to specify the model and to obtain its numerical solution. For the sake of completeness, we provide a brief description of SANs.

A SAN is a Petri net with a few simple extensions. In its simplest form a SAN consists of places and activities (Fig. 1(a)). Each place can have a number of tokens (its *marking*) and each activity moves tokens from one place into another. In the example in Fig. 1(a), `Activity1` moves tokens from `Place1` to `Place2`. An activity can move tokens either at fixed times (deterministic) or at random times (probabilistic). A deterministic activity can be specified using its rate whereas a probabilistic activity is specified using its probability distribution function (PDF). For example, an activity can transfer tokens at the rate of one token per second or it can transfer tokens at random times that are distributed exponentially. Exhaustive search in a space of size N, which is used extensively in this work, has a uniform distribution with a PDF given by:

$$f(x) = \begin{cases} \frac{1}{N} & \text{if } 0 < x < N \\ 0 & \text{otherwise.} \end{cases}$$

In addition to Petri net abstractions, a SAN may also include an *input gate* (Fig. 1(b)). An input gate can define an arbitrary condition (predicate) for enabling an activity. For example, if `Input_Gate1` in the figure has the predicate:

```
if (Place1->Mark() > Place2->Mark())
    return 1;
else
    return 0;
```

then `Activity1` is only enabled when `Place1` has more tokens than `Place2`. Note that `->Mark()` is a pseudo operator that returns the marking of a place.

A SAN may also include an *output gate*. An output gate defines an arbitrary effect after an activity has been enabled (Fig. 1(c)). For example, if `Output_Gate1` in the figure has the function:

```
Place2->Mark()++;
Place3->Mark()++;
Place4->Mark()--;
```

then whenever `Activity1` is enabled, it takes one token from `Place4` and puts one token in `Place2` and one in `Place3`.

4. VPN security model

This section describes the probabilistic model used to analyze VPN security. The model helps quantify the security of a protocol and provides guidance for its secure implementation, configuration and operation. This is important because, as Hills [10] notes, VPN tunnels are misconfigured approximately 90% of the time.

This section explains the details of the SAN model and its parameters. Note that all times in the model are expressed in minutes.

The VPN model comprises two sub-models (atomic models), one models the implementation and configuration of a VPN tunnel, and the other models the VPN environment and operational details. The two sub-models are joined into a composed VPN model using the Rep/Join representation [19].

The first atomic model (*ike*) models the weaknesses of a protocol (Fig. 2). A global variable identifies if the VPN is operating in the main mode or in the aggressive mode. If the VPN is configured in the aggressive mode, an activity models the username enumeration attack. We consider usernames that consist of alphabetic characters and have a length of at most six characters (approximately 309 million possibilities).

Table 1 – Password space size for different complexities.

Password complexity	Space size
6 Characters a–z	3.1E+8
6 Characters a–z, A–Z, 0–9	5.7E+10
8 Characters a–z	2.1E+11
8 Characters a–z, A–Z, 0–9	2.1E+14

If the roundtrip time between the scanner and server is in the order of tens of milliseconds [20] and a window of ten packets is used, then, on the average, it takes 1 ms to check for each username, which implies that 1000 usernames can be checked per second. If the roundtrip time is larger, an appropriate window can be chosen to achieve this rate. With this rate, it is possible to exhaustively scan the username space in approximately 3.5 days. A sophisticated attacker can do better using a fast connection to the server and/or an intelligent guessing algorithm after a username is found. However, in this paper, we consider an unsophisticated attacker in order to obtain an upper bound on VPN security. Note that, because username scanning does not typically cause account lockout, this process is not stopped by the server.

The rate of username enumeration is proportional to the number of system users (more users result in a faster enumeration using exhaustive search). This is modeled by multiplying the base rate (1 per 3.5 days $= 1.94\text{E} - 04/\text{min}$) by the marking of the *usernames* place (i.e., number of tokens in the place). Whenever a username is found, it is moved from the pool of unknown usernames to *usernames_found* using the output gate *username_found*. Note that *usernames_found* is a place that holds the number of usernames enumerated at a given point in time, whereas *username_found* is the output gate that transfers the discovered username from the unknown usernames place (*usernames*) to the *usernames_found* place.

When a username is found, the attacker can start an offline attack to obtain the password. To crack the password, the attacker has to hash different values exhaustively. The cracking speed for MD5 hashes using an AMD Athlon XP 2800+ is around 315,000 attempts per second ($\sim$1.9E + 07 attempts per minute) [10]. Since the cracking speed depends heavily on password complexity, the model is run using different password space sizes. Table 1 shows the password space size for different types of passwords.

The rate of successful attempts is proportional to the number of usernames enumerated, so the rate of the *brute_force* activity is multiplied by the marking of the place *usernames_found*. If a username/password pair is found, then the VPN is breached. Other transactions (e.g., to setup a VPN tunnel after the breach) require negligible time compared with a brute force or username enumeration. As a result, after a username/password pair is found, a token is placed in *vpn_breached*.

The other possible mode of operation is the main mode. As mentioned above, in the main mode, the identities are the IP addresses of the peers. The space of 32-bit IP addresses is approximately fourteen times larger than the space of six-character usernames; thus, the *find_IP* activity that randomly searches the IP address space has a base rate that is fourteen

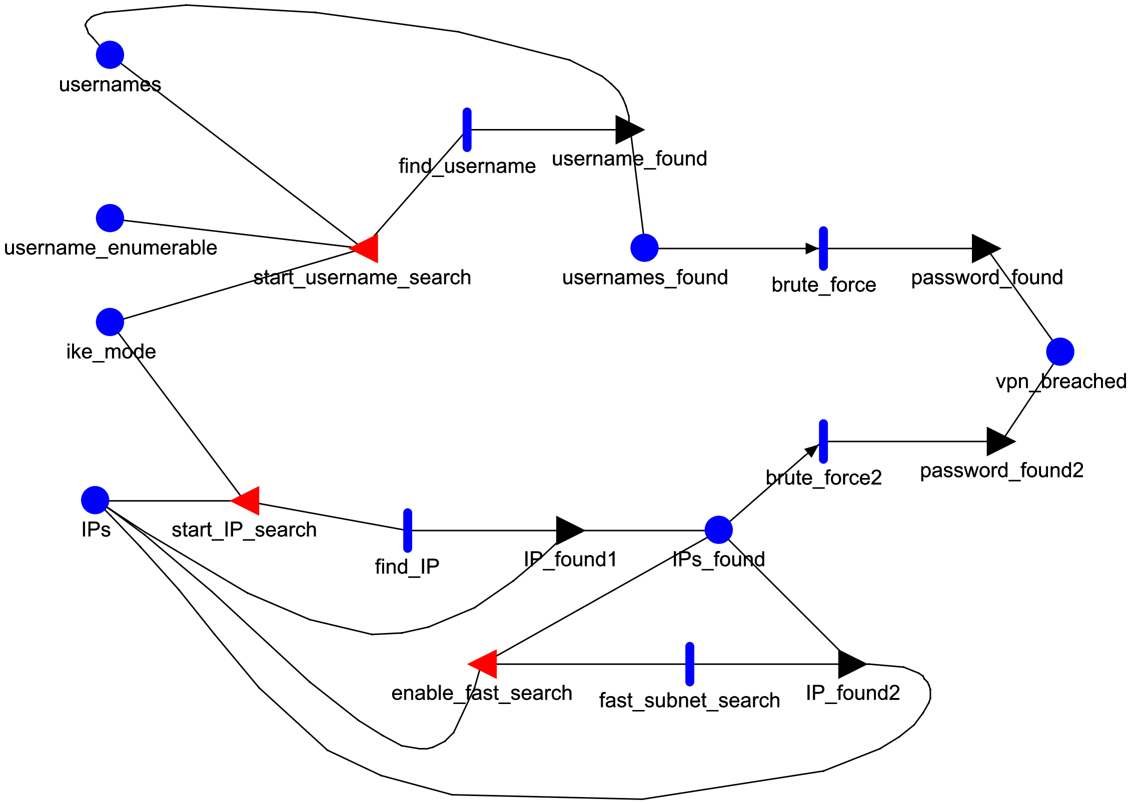


Fig. 2 – Probabilistic model of a VPN.

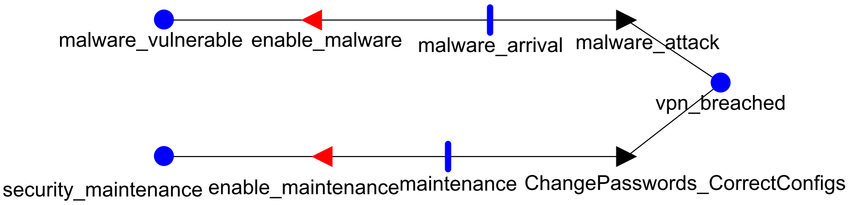


Fig. 3 – VPN malware infection and maintenance models.

times slower than *find_username*. However, upon finding a valid IP address, the attacker can perform a subnet-based search, which speeds up the task of finding other IP addresses (assuming, of course, that most of the clients are in the same subnet). Note that for the main mode to be enabled, at least one IP address must be found by random search. Upon finding a valid IP address, the attacker must exhaustively search the space of PSKs as in the case of the aggressive mode, placing a token in *vpn_breached* whenever an IP/PSK pair is found.

The second atomic SAN model captures the behavior of the VPN environment and its operational maintenance (Fig. 3). VPNs are vulnerable to malware attacks [21]. In particular, malware can modify the configuration of a VPN tunnel in a stealthy manner. Since the VPN tunnel remains operational after the modification, it is difficult for the system administrator to detect such an attack.

We model two environments, one with malware attacks and the other without malware attacks. Malware can maliciously modify a VPN to send packets in plaintext. Therefore, the installation of malware is equivalent to a VPN breach.

The malware infection rate is hard to quantify in industrial systems. For the sake of argument, however, we choose an infection rate of once a month when malware is present. We show later that this rate does not have a significant impact on VPN security.

The activity *malware_arrival* models malware infections. Although malware can also retrieve unsecured passwords, we do not consider it to be a part of the model because it is implementation specific.

VPN maintenance by the system administrator is a preventive and/or corrective action that can improve security. Maintenance involves changing passwords and checking for bad configurations. If a VPN configuration is modified by malware, a maintenance operation can secure the VPN by correcting the configuration and installing a patch that deals with the malware. Also, changing passwords regularly can mitigate exhaustive search attacks, helping secure the

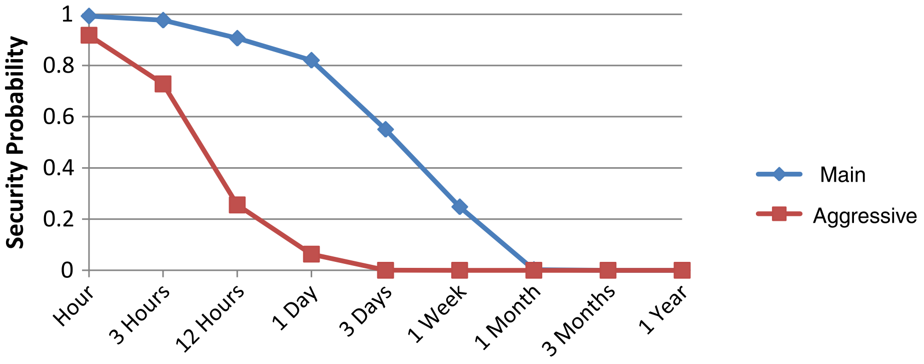


Fig. 4 – Security of aggressive mode versus main mode.

VPN. On the other hand, password changes do not affect username/IP enumeration, so this activity does not flush the *usernames_found* and *IPs_found* places in the model.

5. Experimental results

This section presents the results obtained from simulation experiments using the SAN model. The primary goal was to investigate the probability that the VPN is not in the breached state. In SAN terminology, the reward variable (*security_probability*) is defined as the probability that the marking of the place *VPN_breached* is zero. The value of this probability for each configuration was studied for different time periods: one hour, three hours, twelve hours, one day, three days, one week, one month, three months and one year.

VPN security was also studied for different IKE modes (aggressive vs. main mode); password complexity (Table 1); numbers of users/machines (1, 10, 100 and 250); environments (with and without malware); and maintenance rates (once every week, month, three months, one year and no maintenance). Given the number of factors of interest, a large number of experiments are possible, but not all of them are meaningful. Therefore, we only performed experiments in which a few parameters of interest were changed at a time.

The first experiment compared VPN security in the aggressive mode versus the main mode. The main mode is generally more secure because, in order to perform offline password cracking, the attacker has to conduct a man-in-the-middle attack. Even if this attack is successful, the space of 32-bit IP addresses is larger than the space of usernames. The experiment assumed that the attacker can perform a man-in-the-middle attack; otherwise, the main mode is not vulnerable to the attack (i.e., security probability is one at all times).

The passwords (or PSKs) for both modes were selected from the space of six alphabetic characters (3.1E+08); the system was assumed to have ten users (usernames or IP addresses); security maintenance was not performed; and malware was not present. The results are shown in Fig. 4. Note that the security of a VPN tunnel diminishes over time. However, the security declines faster for the aggressive mode than for the main mode. The aggressive mode is less than 50% secure after six hours whereas the main mode reaches this level after about four days. Note also the short lifetime of a six-character alphabetic password for a VPN tunnel.

The second experiment studied the effect of password complexity on the overall security of a VPN in the aggressive mode. To observe the effect of password complexity alone, maintenance and malware were switched off in the experiment. The system had ten different users. The results are shown in Fig. 5.

As expected, the overall security of a VPN increases with password complexity. Note that eight-character alphanumeric passwords are secure for a much longer period of time, but even this type of password is less than 65% secure after one year. On the other hand, six-character alphanumeric passwords are less than 60% secure after just one day.

The third experiment studied the effect of maintenance frequency on VPN security. The experiment assumed that IKE was used in the aggressive mode, that the system had ten users and that six-character alphanumeric passwords were used. The results in Fig. 6 show that frequent maintenance mitigates the effect of weak configurations. Note that the security probability reduces until it reaches a minimum before any maintenance starts; after this, the probability increases with frequent maintenance. Since the rate of maintenance is higher than the rate of password cracking in each case, the security probability reaches one at steady state. This does not mean that it is impossible to break the VPN tunnel as time passes; rather, it implies that the portion of time that the VPN tunnel is breached diminishes over longer time periods. Note that the declining security trend during the first few days can be repeated after each successful maintenance. The effect is not shown in Fig. 6 because the security probability represents the steady state measure of security at any time.

The fourth experiment studied the effect of malware attacks versus weak VPN passwords. Fig. 7 shows the results obtained for two values of password complexity (six- and eight-character alphanumeric passwords) with and without frequent (once a month) malware attacks. In the experiment, IKE used the aggressive mode and no maintenance was performed. A counterintuitive result from this experiment is that malware infections have little impact on the security of a weakly-configured VPN because the dominant effect in the aggressive mode is imposed by the ability of an attacker to

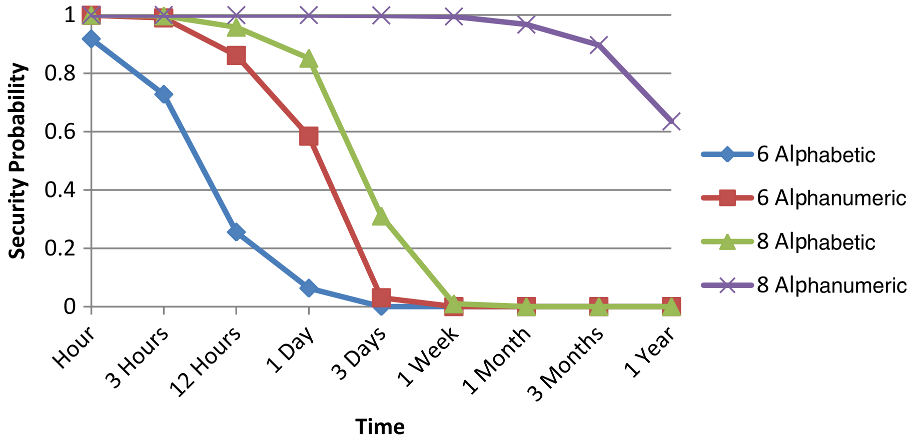


Fig. 5 – Effect of password complexity on VPN security.

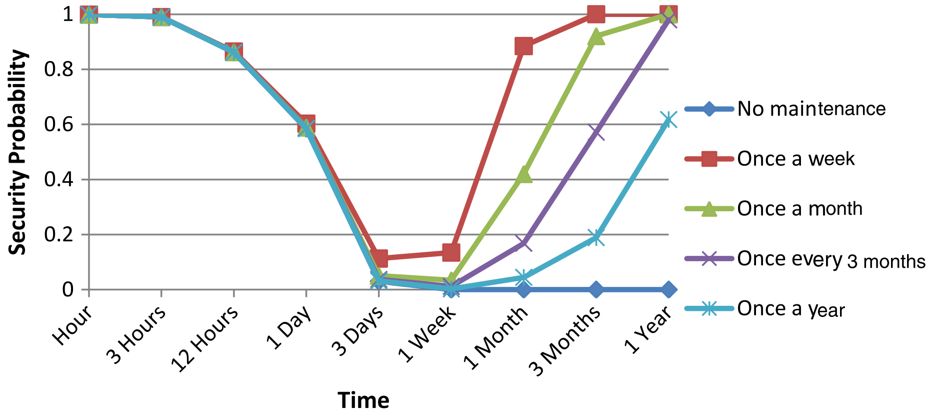


Fig. 6 – Effect of maintenance frequency on VPN security.

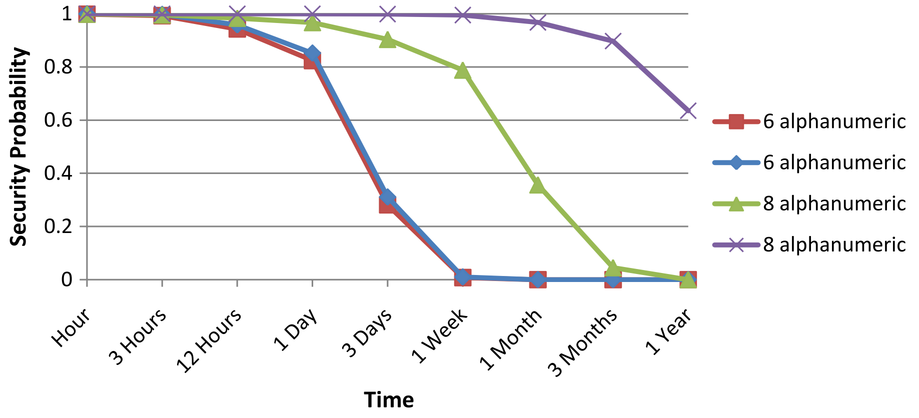


Fig. 7 – Effect of malware on VPN security.

crack a six-character alphanumeric password. On the other hand, in the case of a strong password, frequent malware infections considerably weaken VPN security. Therefore, we can conclude that the impact of a malware infection depends on VPN configuration. If the rate of password cracking is higher than the rate of infection, malware has little impact on the system. As a result, priority must be given to secure VPN configurations. Note that this experiment only considered the effect of malware on the security of a VPN tunnel. Malware infections have other negative security impacts that were not modeled in the experiment.

Next, we studied the effect of the number of users on overall VPN security. As shown in Fig. 8, systems with large user populations are much less secure than systems with

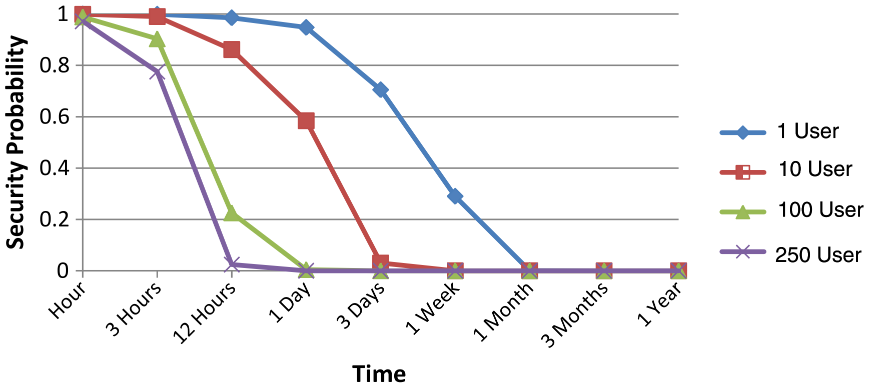


Fig. 8 – Effect of the number of users on VPN security.

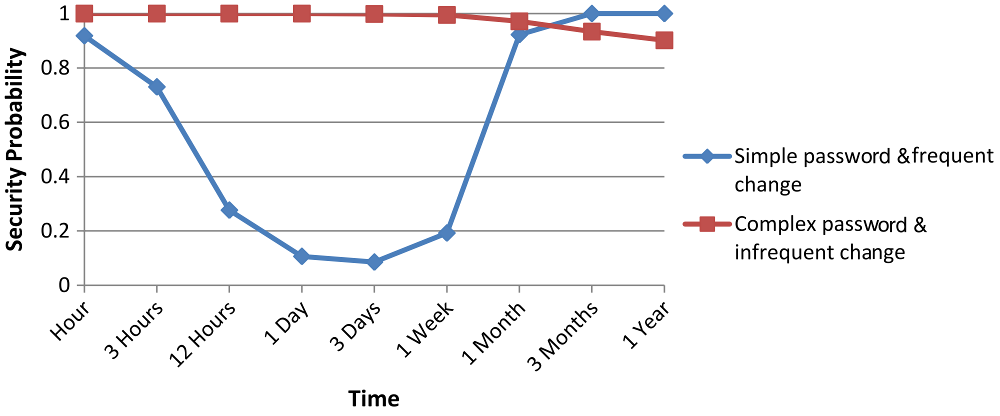


Fig. 9 – Trade-off of password complexity vs. frequent maintenance.

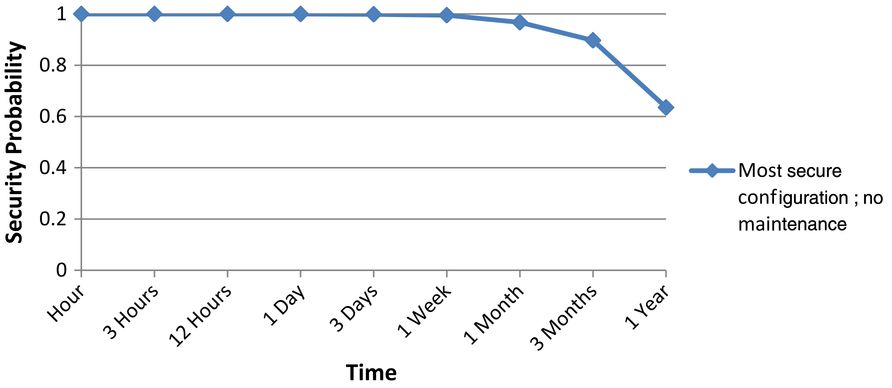


Fig. 10 – Effect of secure parameters with no maintenance.

few users because an attacker has a higher chance of finding valid usernames/passwords (or IPs/PSKs). This experiment assumed that IKE was used in the aggressive mode, that six-character alphanumeric passwords were used, that security maintenance was not performed and that malware was not present.

The sixth experiment was designed to answer an important question: is it better to choose more secure passwords or to perform maintenance more frequently? The experiment considered two systems, one with six-character alphabetic passwords and once-a-week maintenance and the other one with eight-character alphanumeric passwords and maintenance every three months. The results are shown in Fig. 9. Weak passwords with frequent maintenance are less secure in the short term, but after a while (one year) complex passwords begin to expire and the overall security (of the second system) decreases. Note also that changing passwords every week can be a huge administrative burden.

Table 2 – Average security of a VPN over a year with different parameters.

Password space	Average security	Number of users	Average security	Maintenance frequency (1/day)	Average security
3E+8	6E−4	1	1.1E−2	1.4E−1	0.97
6E+10	3E−3	10	2.6E−3	3.3E−2	0.93
2E+11	5E−3	100	5E−4	1.1E−2	0.84
2E+14	7.1E−1	250	3E−4	2.8E−3	0.49

The seventh and final experiment focused on a single configuration: the most secure configuration with no maintenance, complex (eight-character alphanumeric) passwords, ten users and no malware. Fig. 10 presents the results of the experiment. The importance of regular maintenance is clear—even a relatively secure configuration becomes less than 65% secure after one year without proper maintenance.

6. Statistical analysis of significance

The previous section presented experimental data pertaining to the impact of various parameters on VPN security. In order to fully understand the impact of these parameters, we now perform a statistical analysis of significance of the experimental data.

One way to analyze the significance of various parameters on VPN security is regression analysis. In regression analysis, the value of a dependent variable is predicted by changing an independent variable in a regression function. The significance of an independent variable in changing a dependent variable is estimated based on how much the former impacts the latter in a statistical sense.

We use linear regression to analyze the significance of the parameters in a VPN deployment. Linear regression attempts to estimate the dependent variable using a linear function of the independent variable. In the following equation, y linearly depends on x and ε is the error.

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad i = 1, \ldots, n. \tag{1}$$

The value of y is estimated using linear regression by $\hat{y}$ in the following equation:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i. \tag{2}$$

The overall error of estimation is calculated by the sum of squared errors (SSE), which is given by:

$$\mathrm{SSE} = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2. \tag{3}$$

In fact, the linear regression of y is obtained by minimizing the SSE.

A measure of significance is given by the coefficient of determination, R^2, which is defined by:

$$R^2 = 1 - \frac{\sum_i (y_i - \bar{y})^2}{\sum_i (\hat{y}_i - \bar{y})^2} \tag{4}$$

where $\bar{y}$ is the average value of y_i.

Assuming that the VPN is operating in the aggressive mode and no malware is present, we evaluate the significance of user population, password space size and maintenance frequency on VPN security. Table 2 shows the average security of VPN over a year given different parameter values. Note that the data comes from the experiments described in Section 5.

Table 3 – Coefficient of determination for VPN parameters.

Parameter	R^2
Password space	0.999
Number of users	0.413
Maintenance	0.410

Table 3 presents the results of the regression analysis. The results demonstrate that 99.9% of variations in VPN security can be explained linearly by the password length whereas only 41.3% and 41% of the variations are explained by the number of users and the maintenance frequency, respectively. The results can be explained by the fact that password length is directly correlated with the difficulty of password cracking attack, and thus, VPN security. Consequently, password length has a direct, linear impact on VPN security. The other two parameters do not illustrate a direct correlation. For example, maintenance can only change the passwords (not the enumerated usernames), so it only has a significance of 41%. Also, the user population can change the rate of user enumeration, but it cannot affect weak passwords within the system; thus, it only has 41.3% significance.

Clearly, in order to securely deploy a VPN, all the parameters of interest must be chosen safely. However, the statistical analysis shows that password length has the most significant impact on VPN security.

7. Security recommendations

The simulation results provide valuable insights into securing VPNs, especially in industrial control environments where a tunnel is the only means to establish communications security and the tunnel may, therefore, last for a long period of time.

Based on the experimental results, the following recommendations can be made regarding VPN security:

- The aggressive mode for IPSec VPNs provides fast tunnel establishment and less overhead that render the mode an attractive option for industrial environments where timing is critical. However, the mode suffers from serious protocol flaws that can result in security breaches in a relatively short time. Therefore, the aggressive mode should not be used in critical applications. Secure configurations using the main mode and certificate-based authentication provide stronger VPN tunnels at the expense of higher overhead and slower connection establishment.

- Long alphanumeric passwords or PSKs should be used to achieve acceptable security.
- Even with complex passwords, frequent maintenance must be performed to lower the risk of a successful attack, especially when the adversary has significant computational resources. Note that personal "supercomputers" and botnets can significantly reduce the password cracking time.
- A weak configuration can have a dominant effect even when malware infections are frequent. Securely configuring a VPN is the first step to countering attacks.
- Less populated VPNs are more secure. When a VPN has a large number of users, other parameters must be stronger (e.g., longer passwords and frequent maintenance). In the case of VPN tunnels for industrial control applications, it is advisable to keep the number of users as low as possible.
- Usernames (and IP addresses) used in a VPN must be changed or rotated periodically to reduce the risk of username enumeration attacks.

8. Related work

Although probabilistic analysis has been widely used to investigate system reliability, its application to security has not attracted much attention until recently. Wang, et al. [22] have proposed the use of probabilistic models for analyzing system security. They have shown that modeling security using Markov chains can be quite informative and can facilitate the design of secure systems. Singh, et al. [23] have used probabilistic models to study the security of intrusion-tolerant replication systems.

Several previous efforts and standards recommend that VPNs be used to secure industrial control protocols. IEC 62351 [2–4] recommends that VPNs be deployed for protocols such as DNP3 and 61850. Okabe, et al. [5] propose the use of IPSec and KINK to secure non-IP-based control networks. Gungor and Lambert [24] discuss the use of MPLS and IPSec VPNs to provide security for electrical power system applications. Sempere, et al. [25] demonstrate the performance and benefits of using VPNs over IP (among other technologies) for supervisory control and data acquisition (SCADA) systems. Alsiherov and Kim [26] propose the use of IPSec VPNs to ensure integrity, authenticity and confidentiality in SCADA networks; however, they suggest that IPSec be configured in the PSK mode for efficient management. Patel, et al. [6] discuss the use of TLS and IPSec VPNs to wrap SCADA protocols. Alsiherov and Kim [1] suggest using IPSec between SCADA sites to provide security when IEC 62351 is not implemented.

Hills [10] has identified several VPN security flaws and has analyzed the presence of secure configurations in VPN deployments. Hamed, et al. [27] have developed a scheme for modeling and verifying IPSec and VPN security policies. Finally, Baukari and Aljane [28] have specified an auditing architecture for monitoring the security of VPNs.

9. Conclusions

A stochastic model of a VPN and its environment provides a powerful framework for investigating the impact of various configurations and operational modes on VPN security in industrial control environments. Simulations of the model assist in quantifying the security of control protocols and evaluating security trade-offs, thereby providing a basis for the secure deployment of VPNs. The results also provide valuable recommendations for securely configuring VPNs in industrial control environments. Our future research will study other VPN protocols (e.g., TLS and L2TP) and quantify their security properties. Also, we plan to incorporate detailed models of malware infections and man-in-the-middle attacks to study their impact more meticulously. Our future research will also model other industrial control protocols using SANs with the goal of evaluating their benefits and limitations.

REFERENCES

[1] F. Alsiherov, T. Kim, Research trends on secure SCADA network technology and methods, WSEAS Transactions on Systems and Control 8 (5) (2010) 635–645.

[2] International Electrotechnical Commission, Communication Network and System Security—Profiles including TCP/IP, Technical Specification IEC TS 62351-3, Geneva, Switzerland, 2007.

[3] International Electrotechnical Commission, Security for IEC 60870-5 and Derivatives, Technical Specification IEC TS 62351-5, Geneva, Switzerland, 2009.

[4] International Electrotechnical Commission, Security for IEC 61850, Technical Specification IEC TS 62351-6, Geneva, Switzerland, 2007.

[5] N. Okabe, S. Sakane, K. Miyazawa, K. Kamada, A. Inoue, M. Ishiyama, Security architecture for control networks using IPSec and KINK, in: Proceedings of the Symposium on Applications and the Internet, pp. 414–420, 2005.

[6] S. Patel, G. Bhatt, J. Graham, Improving the cyber security of SCADA communication networks, Communications of the ACM 52 (7) (2009) 139–142.

[7] M. Majdalawieh, Security Framework for DNP3 and SCADA, VDM Verlag, Saarbruken, Germany, 2008.

[8] International Electrotechnical Commission, IEC 61850 Standard, Technical Specification IEC TS 61850, Geneva, Switzerland, 2003.

[9] Modbus-IDA, Modbus Application Protocol Specification V.1.1b, Hopkinton, Massachusetts www.modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf, 2006.

[10] R. Hills, Common VPN security flaws, White Paper, NTA Monitor, Rochester, United Kingdom www.nta-monitor.com/posts/2005/01/VPN-Flaws-Whitepaper.pdf, 2005.

[11] R. Brown, Stuxnet worm causes industry concern for security firms, Mass High Tech, Boston, Massachusetts www.masshightech.com/stories/2010/10/18/daily19-Stuxnet-worm-causes-industry-concern-for-security-firms.html, October 19, 2010.

[12] W. Sanders, J. Meyer, Stochastic activity networks: formal definitions and concepts, in: E. Brinksma, H. Hermanns, J. Katoen (Eds.), Lectures on Formal Methods and Performance Analysis (LNCS 2090), Springer Verlag, Berlin–Heidelberg, Germany, 2001, pp. 315–343.

[13] G. Balbo, Introduction to stochastic Petri nets, in: E. Brinksma, H. Hermanns, J. Katoen (Eds.), Lectures on Formal Methods and Performance Analysis (LNCS 2090), Springer-Verlag, Berlin–Heidelberg, Germany, 2001, pp. 84–155.

[14] K. Paterson, A cryptographic tour of the IPSec standards, Information Security Technical Report, vol. 11, (2), pp. 72–81, 2006.

[15] M. Bellare, R. Canetti, H. Krawczyk, Keying hash functions for message authentication, in: Proceedings of the Sixteenth International Cryptology Conference, pp. 1–15, 1996.

[16] D. Harkins, D. Carrel, The internet key exchange (IKE), RFC 2409 (1998).

[17] N. Nayak, S. Ghosh, Different flavors of man-in-the-middle attack: Consequences and feasible solutions, in: Proceedings of the Third IEEE International Conference on Computer Science and Information Technology, pp. 491–495, 2010.

[18] R. Pereira, S. Beaulieu, Extended authentication within ISAKMP/Oakley (XAUTH), Internet Draft (1999).

[19] D. Deavours, G. Clark, T. Courtney, D. Daly, S. Derisavi, J. Doyle, W. Sanders, P. Webster, The Mobius framework and its implementation, IEEE Transactions on Software Engineering 28 (10) (2002) 956–969.

[20] P. Li, W. Zhou, Y. Wang, Getting the real-time precise roundtrip time for stepping stone detection, in: Proceedings of the Fourth International Conference on Network and System Security, pp. 377–382, 2010.

[21] S. Dispensa, How to reduce malware-induced security breaches„ March 31, 2010.

[22] D. Wang, B. Madan, K. Trivedi, Security analysis of SITAR intrusion tolerance system, in: Proceedings of the ACM Workshop on Survivable and Self-Regenerative Systems, pp. 23–32, 2003.

[23] S. Singh, M. Cukier, W. Sanders, Probabilistic validation of an intrusion—tolerant replication system, Proceedings of the International Conference on Dependable Systems and Networks (2003) 615–624.

[24] V. Gungor, F. Lambert, A survey on communication networks for electric system automation, Computer Networks 50 (7) (2006) 877–897.

[25] V. Sempere, T. Albero, J. Silvestre, Analysis of communication alternatives in a heterogeneous network for a supervision and control system, Computer Communications 29 (8) (2006) 1133–1145.

[26] F. Alsiherov, T. Kim, Secure SCADA network technology and methods, in: Proceedings of the Twelfth WSEAS International Conference on Automatic Control, Modeling and Simulation, pp. 434–438, 2010.

[27] H. Hamed, E. Al-Shaer, W. Marrero, Modeling and verification of IPSec and VPN security policies, in: Proceedings of the Thirteenth IEEE International Conference on Network Protocols, pp. 259–278, 2005.

[28] N. Baukari, A. Aljane, Security and auditing of VPN, in: Proceedings of the Third International Workshop on Services in Distributed and Networked Environments, pp. 132–138, 1996.