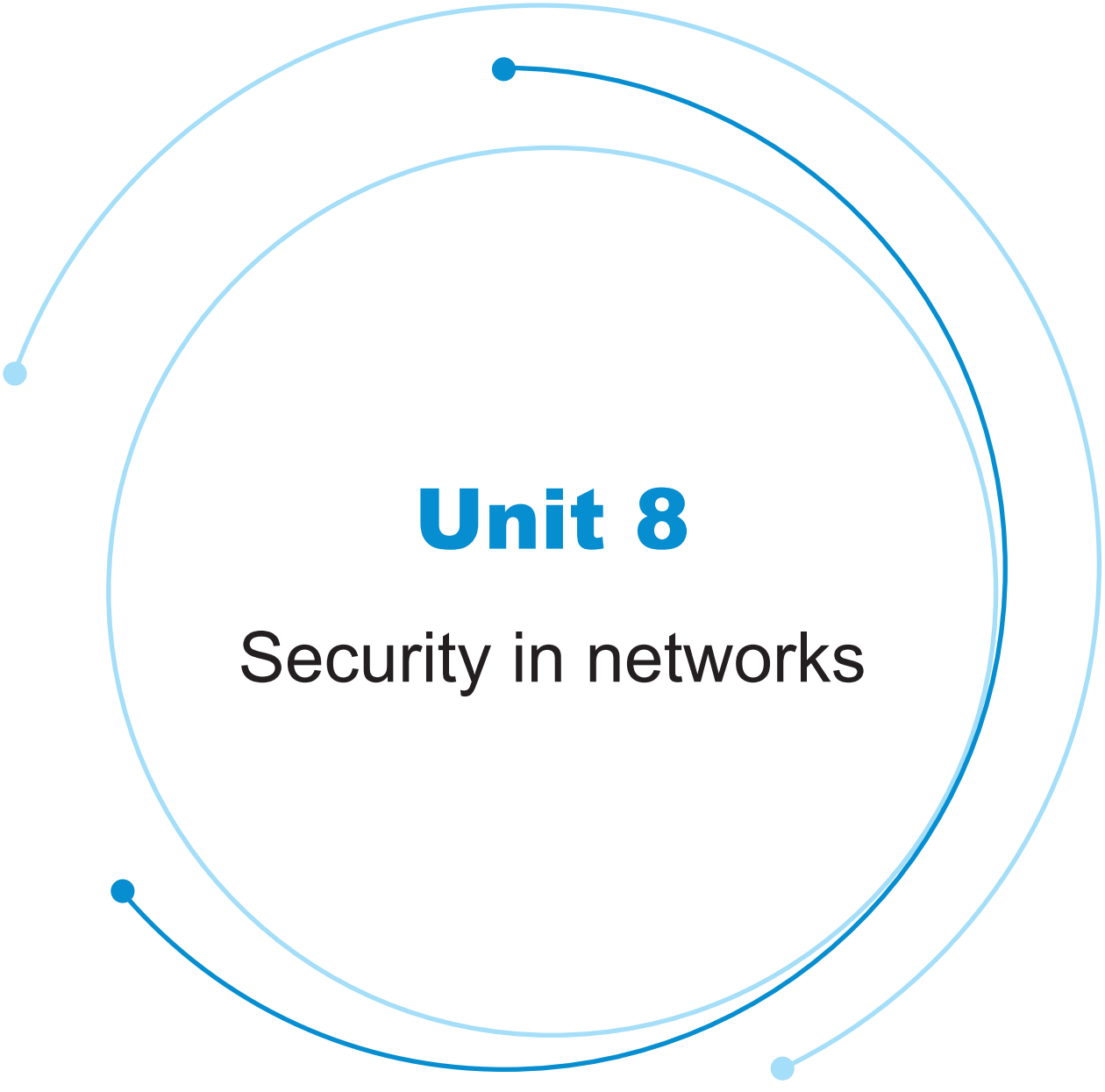
ELEC S212

Network Programming and Design



Unit 8

Security in networks

香港公開大學
THE OPEN UNIVERSITY OF HONG KONG
科技學院 School of Science and Technology

Course team

Developer: Jacky Mak, Consultant

Designer: Ross Vermeer, ETPU

Coordinator: Dr Philip Tsang, OUHK

Member: Dr Steven Choy, OUHK

External Course Assessor

Prof. Cheung Kwok-wai, The Chinese University of Hong Kong

Production

ETPU Publishing Team

The Open University of Hong Kong
Ho Man Tin, Kowloon
Hong Kong

This course material is printed on environmentally friendly paper.

Contents

Overview

Decades ago, mainstream computers were large-scale mainframes, and only authorized users could access them through dedicated terminals. In this monolithic environment, computers posed little threat to the world.

Today's landscape has changed drastically, however. Downsizing of human resources has placed computers in much broader use than ever before. The emergence of the Internet has fueled this proliferation by allowing computers to join up with each other around the globe. Unfortunately, there is also a flip side to such advances: hackers' activities have become far-reaching, and viruses can now spread with unprecedented speed to a vast number of machines.

As a result of these changes, computer security is now an important subject. The aim and focus of such security is to provide proper protection for systems and networks against attacks, exploitation and authorized usage.

We have prepared this unit as a general introduction to computer security, which is a broad subject, but we have put the focus of our attention on network security. First, we introduce to you the fundamentals of computer security. After that, you will learn how to build security features into networks during the design phase, which is more effective than making such features after-thoughts.

Security in the network operating system (NOS) is of no less importance than any other aspects of security, because any fault in the NOS could develop into a threat to all network applications that the NOS is serving. For example, do you remember the security saga caused by the worm 'Code Red' in the early 2000s, in which hundreds of thousands of systems running Windows servers were brought to their knees?

We then go on to discuss cryptography, encryption and Public Key Infrastructure (PKI). These tools provide the blocks upon which various security protocols can address the three fundamental security goals: confidentiality, integrity and authenticity. The unit ends with an introduction to some popular security protocols such as SSL, SSH and SCP, and some authentication protocols such as Kerberos.

In short, this unit:

- discusses the goals and risks in network security;
- describes the implementation of security policies and audits;
- describes common cryptography algorithms;
- describes the mechanisms for authentication and integrity;
- discusses resilient network designs and NOS security issues; and

- identifies and describes common network security controls, including firewalls, proxy servers, remote access and SSL.

You will find this unit challenging simply by virtue of the subject matter it covers. After getting through it, though, you will feel more confident about computer networking since you will be able to deal with it in a more holistic and secure way.

Security goals

When dealing with computer security, we should begin with its end in mind. That is, we should be clear about what good security means to us in first place. What sort of qualities should a secure computing environment exhibit? If we don't get our focus clear in the beginning, we will just wander aimlessly around the vast landscape of computer security, and get lost sooner or later.

Any organization which has addressed security properly should have accomplished the following four security goals: **confidentiality**, **integrity, availability** and **authenticity**.

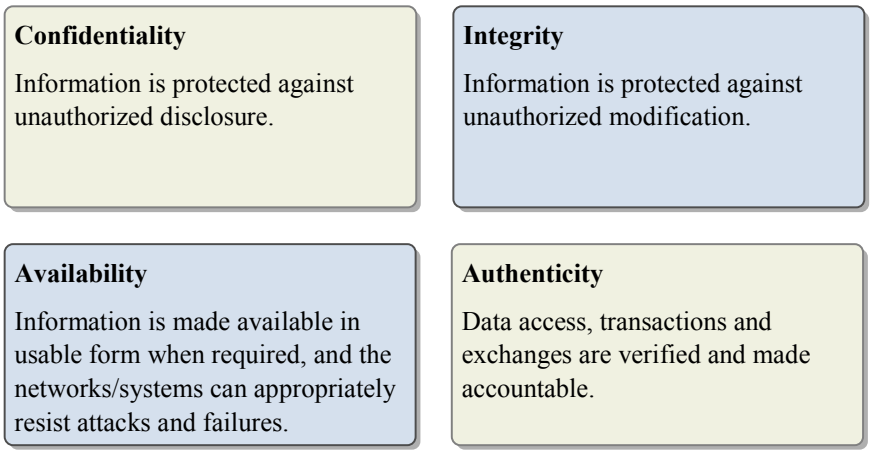


Figure 8.1 An organization's four main security goals

Let's consider each of these goals in a bit more detail:

- *Confidentiality* — Attackers can thwart confidentiality mechanisms by sniffing network traffics to steal passwords or access sensitive data. Encryption, access control, and data classification systems should therefore be put in place to address this goal.

- *Integrity* — Networks often open up a conduit for viruses, Trojan horses or hackers to intrude into and wreak havoc on computer systems, thus compromising the integrity of the data residing therein. Strict access control, network intrusion detection, and checksum/signature verification should be implemented to uphold this goal.

- *Availability* — Both the availability of networks themselves, and that of the computer systems running on them, are important, especially for critical systems that serve users across the globe and round the clock 24/7 (24 hours a day, 7 days a week). Denial of service (DoS) attacks launched remotely by hackers over networks is a popular method of disrupting networks or systems availability. Firewalls, elimination of single points of failure (e.g. dual subscriptions to two different Internet service providers, redundancy mechanisms, etc.) are effective means of counteracting threats to availability.

- *Authenticity* — The identity of an individual should be verified before being granted access to systems. The transactions carried subsequently by that individual should also be tracked. Any specific action that individual has carried out can be tracked, and they can be held accountable for the consequences arising out of such actions. Smart cards, biometrics, and digital signatures are all measures that can step up the level of authenticity for computer or networking systems.

As far as network security is concerned, ensuring integrity and availability are of paramount importance.

Security management practice

Knowing your security goals is the first step. What comes next is to find out how to achieve them. Security management practice, which is an answer to this, entails the following elements:

- Conduct a **risk assessment** to identify judiciously an organization's information assets which are at risk, assess the impacts if such risks are realized, and determine effective countermeasures.
- Develop **policies** that state the specific security goals the organization is committed to pursuing, and prepare guidelines/procedures to guide such endeavors.
- Carry out a **security audit** to determine whether an organization's information assets are at the expected level of protection, which is usually performed by external qualified auditors.

In the coming sections we will briefly discuss risk assessment, security audits and policies.

Security risks

As an old saying goes, life is full of risk. Risk is the possibility that something that can cause damage does really happen. The possibility of bad weather in summer time in Hong Kong is not low. The organizer of an outdoor concert should recognize this risk, understand its impact and then deal with it accordingly. The same concept applies to network security. Below are different types of security risks common to computing environments.

Risks associated with people:

- social engineering or snooping to obtain passwords
- incorrectly creating or configuring user IDs, groups, and their rights on file server
- dishonest or disgruntled employees
- easy-to-guess passwords.

Risks associated with transmission and hardware:

- intrusions into networks through modems attached to network devices
- spying on unprotected sensitive data hosted on the subnet open to the public
- interceptions of unencrypted data transmitted over wireless or public networks
- exploitation of unused hubs, routers, or server ports by hackers.

Risks associated with protocols and software:

- security inadequacies in TCP/IP (e.g. falsified IP addresses, no authentication)
- trust relationships between servers leaving an entire network vulnerable to attack
- security flaws in NOSs allowing unauthorized access to systems
- accepting default options (often not optimal) after OS or application installation
- if NOS allows server operators to exit to a command prompt, intruders could run destructive command-line programs.

Risks associated with Internet access:

- inadequate protection caused by improper firewall configuration
- insecure applications (e.g. Telnets or FTPs) transmitting passwords unencrypted
- virus contagion through Web browsing, instant messaging, emails, etc.

Risk assessment

Risk assessment is the process of identifying risks, quantifying the losses that may resulted if such risks are realized, and determining the effective countermeasures to deal with the risks while factoring in their costs and benefits.

Table 8.1 A list of the risks identified in the risk assessment with their ranking included

Risk descriptions	Potential Losses	Likelihoods	Impacts	Rankings
Leakage of customers' personal information to the Internet via employees' home PCs	High (customer turnover, lawsuits)	High	High	High priority
Network intrusion by hackers	High (database corruption)	High	High	High priority
. . .	. . .	. . .	. . .	. . .
Flood damage	High	Very Low	Medium	Medium priority
File server running out of hard disk space	Medium (delays in jobs)	Medium	Medium	Medium priority
. . .	. . .	. . .	. . .	. . .
System disruption caused by earthquakes	High	Very low	Low	Low priority
. . .	. . .	. . .	. . .	. . .

To determine the impact of a risk, we need to find out the potential loss that the risk may cause, and the likelihood that the risk can be exploited. A high impact risk (e.g. unauthorized database alternations by hackers over corporate networks which are unprotected) is one that can bring about significant loss to an organization and that has a good chance to become a reality. A low impact risk, on the other hand, is one that will lead to few losses or that has almost no chance to be realized. For example, the risk of system disruption caused by earthquake should have a low impact on an organization if it is located in Hong Kong. The identified risks should be ranked according to their impact, as shown in the Table 8.1.

A risk can be dealt with via one of a range of strategies: reduction, assignment, avoidance and acceptance. These are elaborated below:

a Risk reduction methods:

- Install security controls (e.g. smartcard controls, firewalls, anti-virus software, etc.).
- Provide early detection methods (e.g. intrusion detection/prevention systems).
- Raise staff awareness through education.
- Prepare a business continuity plan to reduce the impact on business operations in the event that risks are realized.

b Risk assignment:

- Buy insurance to transfer some or all of the risk.

c Risk avoidance:

- Use alternative means, or do not proceed with the task/operation causing the risk.

d Risk acceptance:

- Live with the risks, especially for those with low impact or the least possibility of being realized.

We have to determine a strategy to deal with each risk. Figure 8.2 depicts the process of doing so. First, we need to find out if we can live with the risk (*acceptance*). If not, we may try to avoid it by using a different way to go about the business (*avoidance*). Or we may try to mitigate the risk by installing appropriate safeguards. We also have to consider transferring the risk to others (*transferral*). We must evaluate the options available, and weigh their costs against their benefits. We then make a choice and lay down a treatment plan to implement our selected options. Ongoing monitoring and review will then maintain their effectiveness.

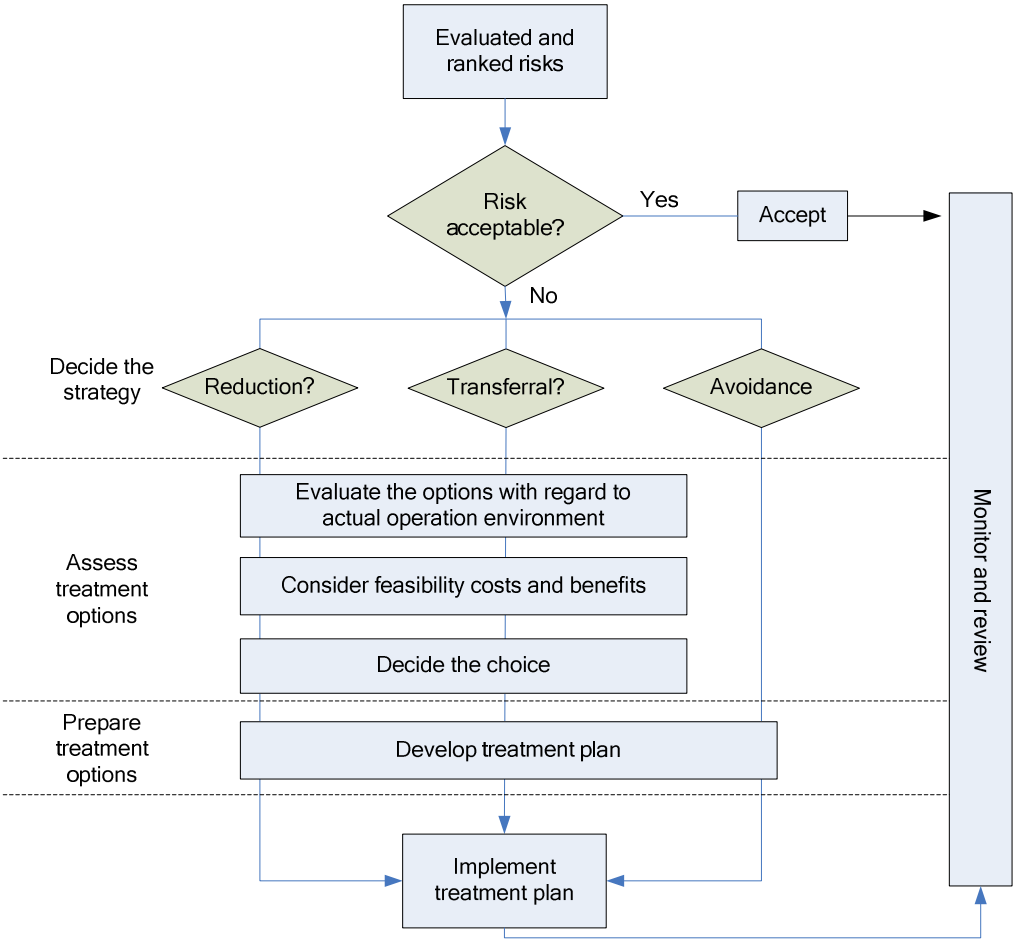


Figure 8.2 The workflow of the risk assessment process

Security policy

The crux of a security program is to protect an organization's computer systems. Risk assessment identifies the risks resided in these systems and determines the impacts that these risks may bring to the organization when they become real. The result of risk assessment helps the organization's management develop a security policy which provides the directions that all security activities should aim at, and expresses the value the management places on the organization's computer security.

A security policy serves as an instrument to help management articulate how much importance they attach to computer security, and their directives for dealing with it. A security policy, which is usually in the form of a comprehensive statement, identifies security goals, risks, level of authority, designated security team, responsibilities for the team. It does not need to go into the details about the equipment or software to be installed; it leaves these details to the security guidelines and procedures that should be developed within company in accordance with the security policy.

Typical goals for security policies:

- Ensure authorized users have appropriate access to resources.
- Prevent unauthorized users from gaining access to network, systems, programs, or data.
- Protect sensitive data from unauthorized access.
- Prevent accidental or intentional damage to hardware or software.
- Create an environment in which network and systems can withstand and recover from any type of threat.
- Communicate each employee's responsibilities.

A committee under the charter of the senior management should be formed which will be responsible for developing the security policy and subsequently communicating it throughout the organization. Committee members should include representatives of the management, information owners, application systems owners, network administrators, end users, etc.

The security policy should adequately cover different areas of security that the organization is concerned with, which may include:

- password management
- software installation
- use of desktop, laptops and loaner machines
- confidential and sensitive data
- network access
- email and Internet use
- remote access
- connecting to remote locations, Internet, and customers' and vendors' networks.

Part of the policy should be a plan for responding to security breaches. It should identify the members of a response team (e.g. dispatcher, manager, technical specialist and public relations specialist) and their roles, responsibilities and duties. Incident-handling procedures should be developed to detail the methods and steps to deal with the security incidents.

Security audit

A security audit is the process of examining the operation of networks and IT systems in great detail in order to determine whether they are as secure as they are supposed to be. As a precursor to security audit, an organization should carry out a risk assessment to determine the risks it's

facing, and decide on countermeasures to deal with them. After those countermeasures are put in place, it can start on a security audit to inspect the effectiveness of the security protection that these and other measures have brought about.

Security audit activities are usually carried out with reference to common security standards such as BS 7799/ISO 17799, Generally Accepted System Security Principles (GASSP), etc. They also include penetration tests to emulate determined hacks to access the hosts and networks which should be protected, and vulnerability scanning tests to scan the hosts and networks for any software loopholes which should have been plugged.

In short, risk assessment focuses on identifying and analysing risks, while the security audit focuses on looking for anything which should have been done but has been left undone.

If your IT department has sufficient skills and time, they can do a security audit themselves. Alternatively, a qualified consulting company can be hired to take up the task. One last very important thing to mention is that a security audit is not a one-off task but should be done on a regular basis, say once every two years.

Reading

Dean (2012) 495–503.

Self-test 8.1

1 What are the key goals of computer security?

2 Table 8.1 sets out five risk examples identified by a risk assessment exercise. For each of these risks, please try to come up with an appropriate strategy to deal it (i.e. choose from possibilities such as *reduction*, *assignment* and *acceptance*) and then determine an appropriate countermeasure that can effectively counteract the risk. Apply both your IT and general knowledge to this exercise, as you would in handling other real-life problems.

3 Is a security policy only concerned with computers or networks? Please explain.

Activity 8.1

1 Complete the first four steps of 'Lab 1.5 — Network Traffic Analysis and Wireless Network Security' from the textbook *A Practical Approach to Internet Programming and Multimedia Technologies*. This exercise will help you realize how easy it is to intercept and analyse the data transmitted in networks.

2 The InfoSec website (資訊安全網) at www.infosec.gov.hk serves as an one-stop portal for the general public to access information and resources on information security, as well as measures and best practices for the prevention of cyber crimes. It includes a wealth of information covering different computer security topics. It is organized to target different groups of computer users including businesses, general users, youngsters and students, etc. In addition to alerts on international security issues, they also provide updates on local security incidents, events and activities.

Take time to visit the website. In particular, please read the information at the following link to consolidate your learning on security management:

http://www.infosec.gov.hk/english/business/security.html

3 The Hong Kong Computer Emergency Response Team Coordination Centre (HKCERT) at http://www.hkcert.org/ is a **CERT** that sets out to provide a centralized contact point for computer and network security incident reporting and response for local enterprises and Internet users who experience security problems. It also disseminates the latest security alerts (e.g. on virus attacks, software vulnerabilities, etc.) and other information related to security issues.

Take time to look around the website. Bookmark the newsfeed link on security alerts feeds at https://www.hkcert.org/getrss/zh/securitybulletin, and visit it regularly in order to keep yourself abreast of the latest security alerts.

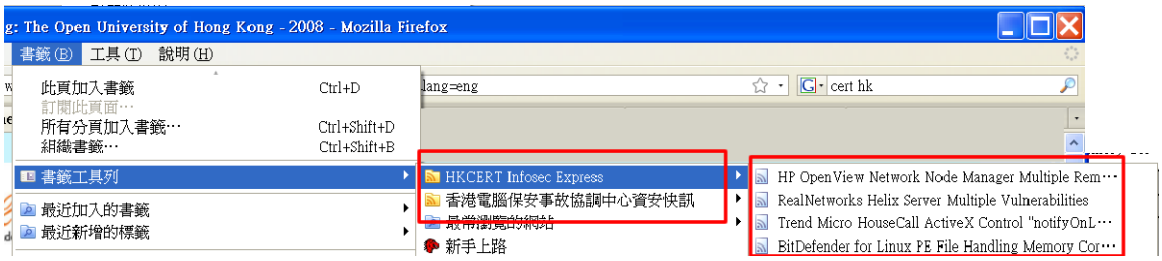


Figure 8.3 HKCERT's latest security alerts

Security in network design

We now go on to discuss some of the tactics and measures that can be used to counteract the threats identified at the risk assessment stage in order to minimize their impacts to an acceptable level. To start with, we will cover the measures that should be implanted into a network during its design phase, thereby building them into the network infrastructure to provide secure protection *centrally* to the whole network. Later in this unit, you will study another section title 'Network Security Protocols' that covers the security protocols suitable for protecting different network applications *specifically*.

Network segmentation

Security should be inbuilt into the design of a network; it should not be an afterthought that tries to patch up a network after it has been built.

A contemporary network design approach is to segregate networks into a number of smaller segments, with each segment serving a particular purpose. For a simple example, we can use internal segments to house the systems (e.g. inventory system, payroll system, etc.) that only internal users can use, and outside segments, also known as the **demilitarized zone (DMZ)** when the Internet is involved, to house the systems (e.g. Web, mail, proxy servers, etc.) that external users will access. Appropriate technical controls such as firewalls, anti-virus scanners and intrusion-detection systems should be installed in the boundary of these segments as well as inside these segments to provide layered protection to the network. This design is depicted in the figure below.

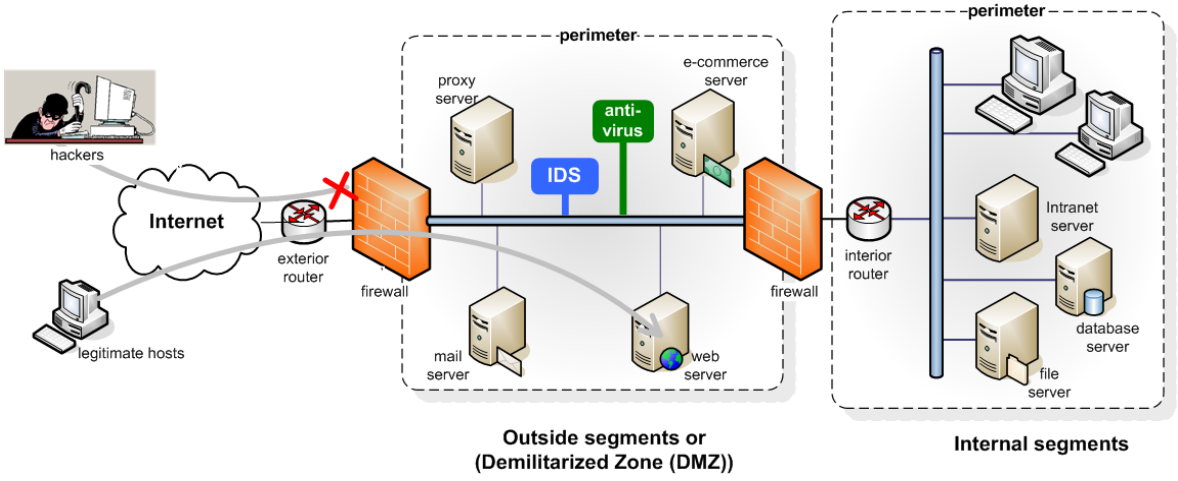


Figure 8.4 A secure network design typifying use of network segmentation

As you can see from the above figure, we use routers to segregate network segments from each other to restrict broadcast packets from going out of its local network segments. If such broadcast information falls into hackers' hands, they could be used for launching attacks. In addition to the routers, we install a pair of firewalls at the boundaries

(this is also known as back-to-back firewall design) to filter out any unauthorized traffic to avoid its passing in. The firewall near the *exterior router* provides the first layer of defence against external attacks. Other technical controls such as an **intrusion detection system (IDS)** and real time anti-virus monitoring tools are also put in place in the outside segments to detect any early signs of invasion, i.e. before substantial attacks can arise.

But what if an attack really occurs? In such cases, the firewall near the *exterior router* provides the first layer of defence. If it turns out that the firewall is compromised by the hacker, the systems that reside in the outside segments will be in jeopardy. But, at that moment, the second layer of defence mounted by the firewall near the *interior router* will kick in to protect the internal systems, which usually hold the most valuable data and applications.

This segmented network design plus the layered defence increases the network's robustness against attacks. It could grant the organization's IT department more time — though it may still not be much time — to identify the source of attacks, contain the damage and plug the loopholes in order to resume normal network operations.

It is important to draw the perimeters of the network segments clearly; to determine which systems should be put under them; and to put in place appropriate defence mechanisms (e.g. firewalls, IDS, anti-virus, etc.) into each perimeter to protect the systems underneath. Some of these defence mechanisms are discussed below.

Firewalls

A **firewall** is a specialized device or a computer installed with specialized software that selectively filters or blocks traffic between networks in order to protect the systems operating within the perimeters of an organization. Firewalls usually reside between two interconnected networks and screen out unwanted traffic from moving from one network to another according to predefined requirements.

One typical type of firewalls is a *packet filtering firewall*, which operates at the network layer of the TCP/IP model. A packet-filtering firewall examines the header of every packet of data received and does the filtering based on predefined criteria such as the following:

- Is there a matched source IP address, or destination IP address, or both?
- Is there a matched source port, or destination port, or both?
- Are required flags set in the IP header?
- Are there matched protocols (TCP, UDP or ICMP)?
- Is this the first packet in new data stream?
- Is it inbound or outbound traffic?

Figure 8.5 elaborates the basic operation of a packet filtering firewall to protect the **Virtual Private Network (VPN)** connection formed between two sites, at Houston and Denver in the US, respectively.

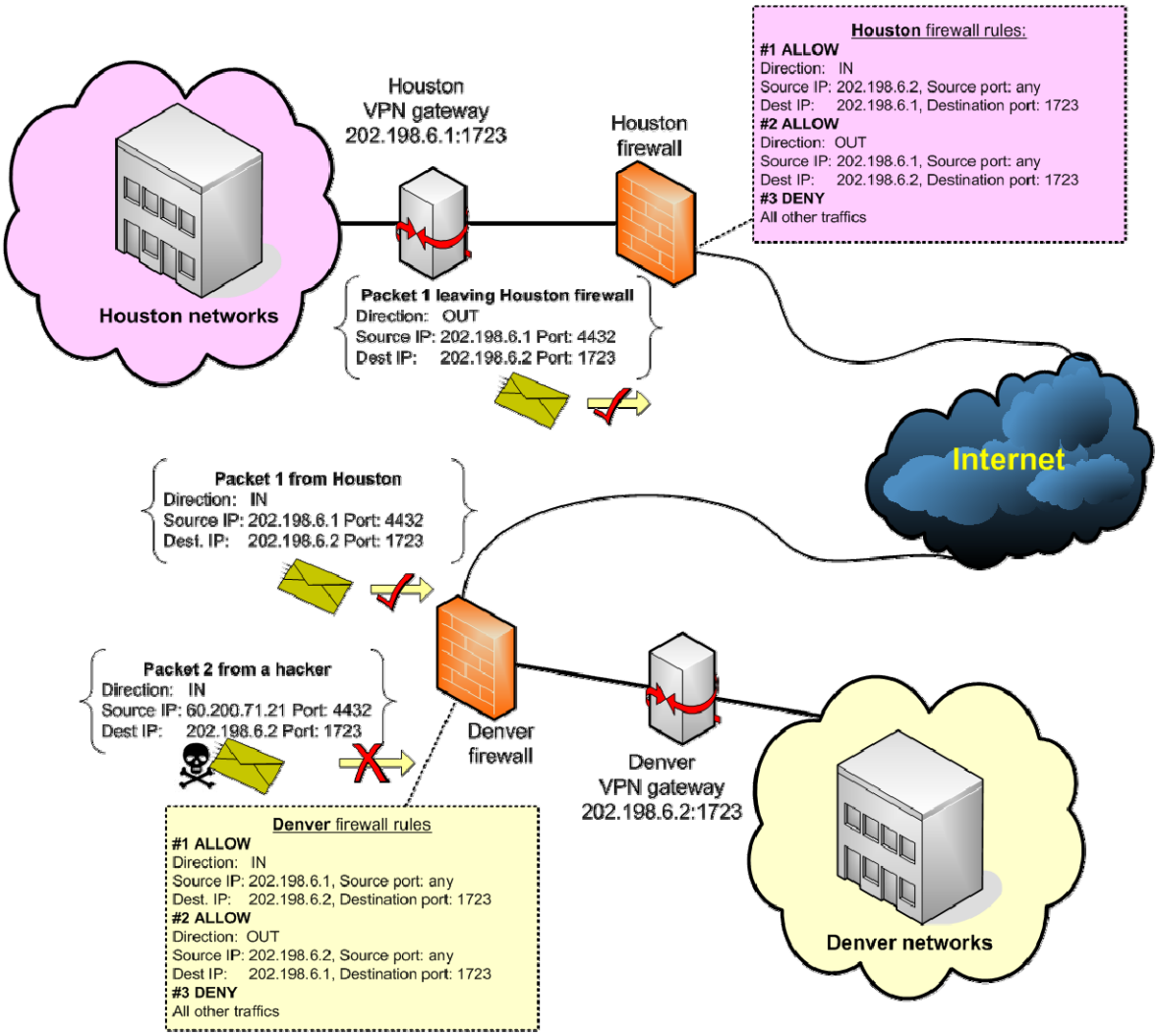


Figure 8.5 Operation of firewall devices

Packet 1 is first generated by the Houston VPN gateway to establish a connection with its Denver VPN gateway counterpart. The Houston firewall inspects the header of the packet, including its direction, source IP address and port, and destination IP address and port, and finds that firewall rule #2 is matched. The packet is therefore allowed to go out past the firewall into the Internet.

Packet 1 then arrives at the Denver firewall, which inspects the header of the packet and finds that firewall rule #1 is matched. The packet is therefore allowed to go through the firewall into the Denver's VPN gateway and then into Denver's networks.

Later on, packet 2 arrives at the Denver firewall, which is possibly created by a hacker. The firewall inspects the header of packet 2 but finds that rules #1 and #2 are NOT matched. Thus, rule # 3 is applied to deny packet 2. In effect, this eliminates the risk that Denver's networks will be attacked by a hacker.

A firewall's rules have to be fine-tuned based on the actual environment they are operating in. Revisions are also needed when the configuration of networks or systems are changed.

There is another type of firewall that can operate at the application layer in addition to the network layer. They are called *content filtering firewalls*. These firewalls can look into the traffic to determine whether their content conforms to the corresponding application protocols (e.g. http), and whether they contain any malicious code (e.g. Trojan code buried in a webpage returned from a malicious host).

In addition, firewalls often come with the **network address translation (NAT)** function which allows a number of internal IP address (e.g. 10.X.Y.Z, 172.168.Y.Z) to share one or a few global IP addresses. Not only can this support more economical use of global IP addresses, which are faced with the problem of depletion, but it can also hide internal computers from the Internet to lower their risk of being attacked.

Proxy servers

A **proxy server** is software that acts as an intermediary between external and internal networks, screening all incoming and ongoing traffic. It is often used in conjunction with firewalls to strengthen the protection of a network. Figure 8.6 shows a proxy server working with a packet-filtering firewall to protect a private/internal network.

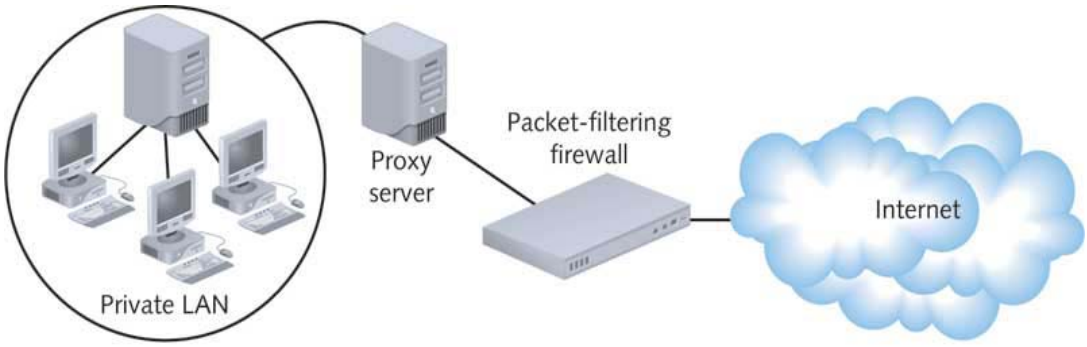


Figure 8.6 A proxy server working with a packet-filtering firewall (Dean 2004, Figure 14-4)

Proxy servers operate at the application layer of the TCP/IP model. One common use is as a Web proxy server. When a user requests a webpage from an external Web server, the Web proxy server will make the request on the user's behalf and then pass back the obtained webpage to the user. As you can see from Figure 8.6, the proxy server is placed in the DMZ. Only its address is exposed to the outside, while the user's address can be kept separate. This can effectively protect the user from external attacks. The proxy server will also save the retrieved webpage in its cache. When another user requests the same page next time, the proxy server can return the cached page to the user immediately, thus speeding up the response.

Remote access

Remote access provides users with convenience since they access the internal systems of their company as if they are in their office physically. However, this kind of remote access must be built into the design of the network in order to ensure that it is covered by adequate protection. Remember any abuse or exploitation of remote access could be catastrophic, since invasion of internal office systems by hackers could result. As mentioned earlier in the 'Firewalls' section, the most valuable data and application often reside in internal systems.

A typical way to implement remote access is to use remote control software, which should support strict access controls including, for example, the following:

- username and password requirement for **authentication**
- application of two-factor authentication (e.g. smartcards, one-time password tokens)
- ability of the host system to call back (if dial-up networking is used)
- support for data encryption on transmission.

VPNs (Virtual Private Networks) provide one of the most popular means for providing remote access. We will discuss VPNs in a later section.

Anti-virus protection

A virus is a program that spreads by attaching itself to other programs. Today, viruses usually spread through emails that arrive with an infected attachment. When a virus's payload executes, it may also do harm to the host system in which it resides (e.g. delete all files).

A Trojan horse is a program that spreads copies of itself through the Internet or local area networks without requiring a host program or system. Some Trojan horses do not need any interaction from a user but can directly infect a network host by exploiting a security vulnerability found in the host. There were some notorious Trojan horses such as CodeRed, Witty, etc. that have led to severe security problems in recent years.

We should set up anti-virus scanners at network servers and client workstations to protect a network against viruses and Trojan horses. One very important thing about anti-virus protection is the need to update the virus signature of the scanners regularly. Using an outdated signature renders the scanners ineffective since new viruses / Trojan horses come out every day.

In addition to scanners, defining appropriate security policies is also important in combating virus threats. We use a security policy to require users' to ensure their anti-scanner always up and running, and to refrain

from installing any program obtained from doubtful origins (e.g. software downloaded unscrupulously from the Internet). Also, they should patch up their computers with the latest security updates to plug any known security vulnerabilities.

Intrusion detection systems (IDSs)

Maintaining a network's security has never been a simple task. It involves ongoing system operation and support to oversee the prevention, detection, response and escalation of abnormal or suspicious activities. An intrusion detection system (IDS) is one of the most useful tools for working out this strenuous task. We install an IDS at strategic locations (e.g. within the DMZ as shown in Figure 8.4) and operate it continuously to look out for any suspicious activity arising from the network. The detection mechanism of the IDS will use signature files to detect the patterns of suspicious attacks. Regular updating of signature files is therefore again important, just as in the case of anti-virus software. When the symptoms of attacks are detected, an IDS will alert the system administrators accordingly, who will in turn take over the matter.

There are two types of IDSs: network-based IDSs and host-based IDSs. The former is to examine network packets transmitted in the network, while the latter is to detect any abnormalities arising in a host system (e.g. unexpected changes to system files). It is common to use IDS tools, both network-based and host-based, to protect Internet Web servers or mail servers.

VPNs

A Virtual Private Network (VPN) is a secure network connection established over an existing public or open network, which is assumed to be unsecure. It uses encryption and authentication technologies to protect the data transmitted through it. Currently, VPN over the Internet is popular for connecting two sites together (i.e. site-to-site connection) or for allowing a remote user to connect back to her main office network environment (i.e. host-to-site connection). A VPN offers a compelling alternative to the dedicated point-to-point links (e.g. T1) for linking up different sites, especially when factoring in its affordable running cost (merely an Internet subscription is needed).

Tunneling

The key concept behind VPN is tunneling, which is the process of encapsulating one type of packet inside another to facilitate some sort of transport advantage. As a result, a communication channel (tunnel) is formed between two end points. This is illustrated in the figure below.

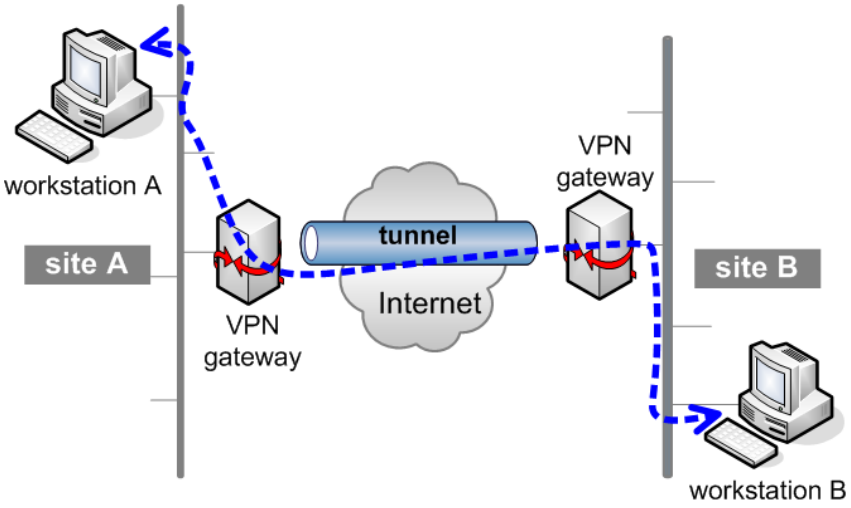


Figure 8.7 A site-to-site VPN channel

A site-to-site VPN channel is formed between two end points (i.e. the VPN gateways in the above figure) with each located at a different site. At the sending end (site A), the original data content from the source workstation A is encrypted by the VPN gateway there, and then put into transportation over the Internet. At the receiving end (site B), the data content is decrypted and restored to its original form by another VPN gateway there, and then forwarded to the destination workstation B. If the data is intercepted during its transit across the Internet, the intruder can hardly make any sense of it since it is encrypted into some unintelligible form. Sophisticated VPN systems impose authentication in their implementation by signing every packet with a secure hash so that the recipient can prove that it originated from a legitimate source.

Different VPN systems carry out the encryption at different layers of the TCP/IP model; this is purely a choice of implementation. Some examples are listed below.

Application layer:	SSH port-forwarding, application proxy
Transport layer:	VPN over SSL
Network layer:	IPSec
Data link layer:	L2TP (Layer 2 Tunneling Protocol), PPTP (Point-to-point Tunneling Protocol)

Among all the security measures we've discussed, the 'VPN over SSL' approach has become increasingly popular, probably due to its simplicity of implementation. VPN systems often come in the form of hardware appliances to be installed as networking equipment to form part of the network itself. In the figure below, the Juniper VPN over SSL solution is used to illustrate a scenario in which a home user connects back to his office network to carry out some urgent support task just as if he were in his office. Please read the numbered textboxes in sequence for the explanation.

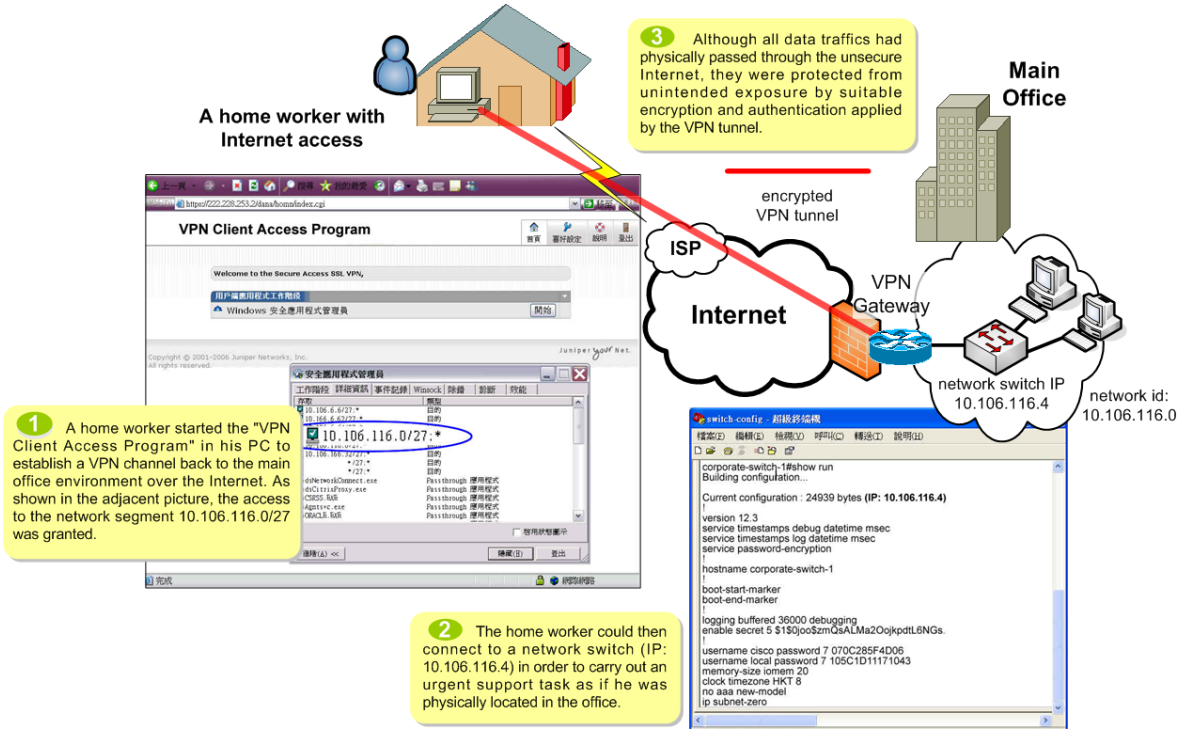


Figure 8.8 A remote host connects back to corporate network using VPN

The OpenVPN (http://www.openvpn.org/) is a popular software-based VPN over SSL solution. OpenVPN is free to use, which makes it a widespread choice among small and medium enterprises, which are wary of their budgets.

OpenVPN employs a lightweight design that can shed quite a few of the complexities that are found in other earlier VPN implementations. Even better, it is free to use. If you would like to better understand or evaluate this useful security tool for your own interest, you can get started with the following reading. The first 25 pages introduce the historic developments of VPN since its emergence in the mid-1990s. It then explains the limitations of the 'VPN over IPSec' implementation approach, and how they are overcome by the 'VPN over SSL' approach, which has become increasingly popular.

Reading 8.1 (online)(optional)

'Understanding the user-space VPN: History, conceptual foundations, and practical usage':

https://openvpn.net/papers/BLUG-talk/BLUG-talk.ppt

At the HowStuffWorks website you can gain a solid understanding of VPNs, and learn about basic VPN components, technologies, tunneling and security. In section 14 on 'Tunneling,' an animation illustrates its implementation.

Reading 8.2 (online)

HowStuffWorks, 'How virtual private networks work':

http://www.howstuffworks.com/vpn.htm

IPSec

Internet Protocol Security (IPSec) defines encryption, authentication, and key management for TCP/IP transmissions. It is an enhancement to IPv4, while it is a native feature of the newer IPv6. IPSec operates at the network layer and, as a result, its protection (e.g. data encryption, authentication, etc.) can go to all applications running on upper layers, which otherwise have to mount such protection for themselves. By contrast, SSL is an application layer protocol. Applications have to make necessary modifications themselves in order to utilize the protection of the SSL protocol.

IPSec has two operation modes:

- The transport mode encrypts only the payload of the IP packets, and is used to protect an end-to-end conversation between two hosts. Since the source and destination IP addresses of the original packet will remain in the header of IPSec packets throughout the route, this mode is not a tunneling protocol, and has little to do with a traditional VPN.

- The tunnel mode encrypts both the IP header and payload of the packets. This allows the source and destination addresses to be different from those of the encompassing packet. This allows the formation of a tunnel. Tunnel mode is more typically used between gateways (routers, firewalls, or standalone VPN devices) to provide VPN functionality.

IPSec can be flexibly deployed to the following scenarios:

- host-based packet filtering to eliminate unwanted traffic, working in ways similar to a firewall

- securing an application by, for example, allowing the access by specific hosts only

- end-to-end security between specific hosts (e.g. client-to-server, server-to-server, and client-to-client), allowing the identities of the hosts to be authenticated and the data to be encrypted

- forming host-to-site (or client-to-gateway) and site-to-site (or gateway-to-gateway) secure tunnels for VPN connection.

Flexibility often comes at the price of complexity; IPSec is no exception. For instance, IPSec requires operating system support since most OS kernels don't allow direct manipulation of IP headers. It turns out that different OSs have different IPSec implementations, which results in substantial interoperability problems among different systems. As a result, the use of IPSec is as of yet not widespread. Currently, the most common type of IPSec implementation is found in some routers or dedicated VPN gateway appliances. These hardware devices are preconfigured with an IPSec configuration when they come out of the factory, so administrators can do little to change them. But by sacrificing some flexibility, they become less complex and thus more practical for real life implementation.

The following optional reading that helps you understand IPSec. It features some good graphical illustrations, which makes IPSec easier to comprehend.

Reading 8.3 (online)(optional)

Friedl, S, 'An illustrated guide to IPsec',

http://www.unixwiz.net/techtips/iguide-ipsec.html

Reading

Dean (2012) 505–14.

You should now be ready to tackle the following self-test.

Self-test 8.2

1 In the context of the firewall configuration in Figure 8.5, let's say a new VPN gateway (IP: 202.198.6.3) at San Jose will be set up to connect to Denver as part of the company's expansion plan. Please modify Denver's firewall rules table, as already provided in the figure, in order to accommodate this expansion. (*Hints*: (i) Insert two rules after rule #2 to allow traffic between San Jose and Denver; (ii) Change rules #3 to #5).

2 With reference to TCP/IP, on which layer does a proxy server operate?

3 What is the major advantage of VPN as compared with other alternative such as dedicated point-to-point connection (e.g. T1 links)?

4 What are the two operation modes of IPSec?

Activity 8.2

1 Through the Internet or by other means, investigate ways to determine whether an IPSec packet is operating in transport or tunnel mode.

2 Download and try out the ClamAV anti-virus software at www.clamwin.com. ClamAV is free to use Open Source Software (OSS) for Windows desktops.

3 Are there any proxy servers at the OUHK? If there are any, what are their host names?

Fault-tolerance in network design

One of the key areas that network security is concerned with is availability, which is meant to ensure that users can use a network anytime during its operating hours to access the data and services that reside there. A fault-tolerant network is one that can carry on its operations even when faults or disruptions arise, and that can recover from such mishaps without causing major impacts on users.

Security threats do not always come from outside. A hardware fault or a software bug can also bring down a network if there is no inbuilt fault-tolerance. The result could be no different from that of external attacks. It is therefore important to design a fault-tolerant network that can continue performing despite unexpected hardware or software errors.

No equipment can go running without a power supply. A redundant power supply (e.g. through uninterrupted power supply (UPS) equipment) is therefore essential to ensure non-stop network operation. Eliminating single points of failure from a network design also improves its robustness. For instance, by using dual network links to connect critical sites together or to connect to the Internet so that even when one link is down, we can divert traffic from one to the other. Also, regular data backups, especially for critical areas such as configuration files and logs, are indispensible for recovery and troubleshooting.

Please work though the following reading to better understand these fault-tolerance measures. Try to apply them when you design or evaluate a network next time.

Reading

Dean (2012) 652–74.

Self-test 8.3

Explain each of these three types of backup operation: full backup, incremental backup and differential backup.

Why does a backup strategy usually include more than one type of backup operation?

NOS security

In *Unit 4* you were introduced to the network operating system (NOS). One of the main functions of an NOS is to manage the services and resources that reside in the network to ensure that they are used by authorized users only, and in the ways that they are designed to be used. An NOS should also provide a management facility that allows administrators to prescribe which users can use which resources, and in what ways (e.g. during what time of a day, which day of week, what kinds of operations, etc.). Quite often an NOS supports the creation of groups of users and assignment of access privileges to these groups in additional to individual users to provide greater flexibility.

In the following sections we will discuss some typical NOS security functions that can contribute to overall network protection.

Logon restrictions

Administrators may impose restrictions on the ways that users log onto network servers. For instance, in a five-day working environment, there should be a ban on server logons for general users during weekends. As a result, in case a hacker has managed to sneak into the office on a Sunday, he still can't access the network servers because the ban is in force. Restricting logons to designated IP addresses and limiting the number of unsuccessful logon attempts are also effective measures to strengthen security.

Passwords

Using passwords is the most common way to authenticate users' identities and to authorize their use of the network services they are entitled to. Ironically, a password is not considered to be a very secure measure, especially when not used properly. Managing passwords is important to avoid the false sense of security that could otherwise result. Some common dos and don'ts on password management include:

- *Do* use a password with a reasonable length (e.g. at least eight characters) and with a mix of any two of letters, integers and symbols.
- *Do* change to new passwords regularly (e.g. at least once every 90 days).
- *Don't* use easy-to-guess passwords (e.g. 'password', your name, etc.).
- *Don't* stick your passwords on your notice board or anywhere others can easily see.
- *Don't* enter your password in a public or shared workstation unless you have been assured of the workstation's security.

Logging

Logging refers to the mechanism to record the activities that have taken place in a system. The logging function of an NOS allows administrators to monitor the usage of network resources; to look out for any potential problems in their early stages, i.e. before they turn into real problems for the network; and to troubleshoot the root causes when problems do really arise. Typical activities that should be logged include:

- user accounts logon/logout time
- unsuccessful logon attempts
- attempts of unauthorized access to network resources
- attempts of unauthorized changes to system settings
- details of remote logon sessions.

Log information should be reviewed regularly. Otherwise security problems may go on developing until they have caused real damage to the network.

Built-in firewalls

You learned about firewalls in an earlier section. There we focused on hardware-based firewall appliances that can safeguard the network perimeters in a corporate network environment. In fact, an NOS also comes with software-based firewalls, which can add an additional layer of protection to a network. Administrators can use these built-in firewalls to define specific rules to block out unwanted traffic from getting into the server system where the NOS is operating. These specific rules may not be suitable for applying to the perimeter firewalls since so doing will in effect apply the rules to all other systems that reside within the perimeter, even if these rules are not suitable for them.

> ***Reading***
>
> Dean (2012) 514–15.

Activity 8.3

Complete Project 14-3 in the textbook *Network+ Guide* by Dean to configure the logging facility of the built-in firewall of your Windows desktop. Inspect the log file pfirewall.log afterwards. This firewall can perform any of three actions: Open, Close and Drop against the connection associated with a packet. Analyse the log file, and see how these three actions are executed.

Cryptography

Cryptography refers to the methods of encoding data in a form that only those it is intended for can read and process (i.e. **encryption**). With some prior knowledge (just like a key), one can easily restore the encoded data to its original form (i.e. decryption).

Encryption basics

Suppose that Alice wants to send a message to Bob. Alice's message in its original form — '`This is a test.`' — is known as the plaintext. Alice encrypts her plaintext message using an encryption algorithm so that the encrypted message, known as **ciphertext**, looks meaningless to anyone else, so far as possible. The plaintext is '`This is a test.`' and the encrypted text becomes '`.tset a si ishT`'. The sequence of character string is the reverse of the original.

In reality, encryption uses more complex mathematical algorithms to scramble data into unintelligible forms. These algorithms, known as **ciphers**, use a key (usually consisting of a train of '1' and '0' bits) to determine how an encryption process should actually take place. We must have prior knowledge of the key in order to decrypt the ciphertext back to the plaintext. Without that, a great deal amount of effort has to be spent to break the ciphertext with a brute-force attack.

Different encryption algorithms offer different degrees of protection, depending on how hard they are to break. The key factors include the sophistication of the algorithm itself, and the length of the key used for encryption. The more sophisticated the encryption algorithm, and the longer the key, the more resistant to breaking the resulted ciphertext will be.

Mind you, however: most encryption algorithms are *not* unbreakable. But don't panic, because what really matters is how much effort or time is required to break them. If the cost required to break an algorithm is greater than the value of the encrypted data or the time required to do so is longer than the time the data needs to remain secret, you are probably safe!

Cryptography is instrumental in addressing the fundamental security goals of confidentiality, integrity and authenticity, which are illustrated below:

Table 8.2 Confidentiality, integrity and authenticity

Confidentiality	Encrypted data can only be viewed by intended recipients who have the decryption key in hand. Any other interceptor cannot make any sense of it.
Authenticity	Encryption can help ensure that the data is truly issued by the stated sender and has not been forged by an intruder.
Integrity	Working in conjunction with the hash functions (to be explained later), encryption can ensure the transmitted data is tamper-proof.

There are two fundamental types of encryption: symmetric and asymmetric, which are discussed as follows.

Symmetric encryption

Symmetric key encryption is also known as private encryption, because it uses the same key and same cryptographic algorithm to encrypt and decrypt data. All sorts of cryptographic algorithms involve substituting one thing for another. They take a piece of plaintext and then compute and substitute its content to create the encrypted message. A very old simple symmetric key algorithm is known as the Caesar cipher (a cipher is a method for encrypting data). For English text, the Caesar cipher works by taking each letter in the plaintext message and substituting the letter in k letters later (allowing wrap-around; i.e. having the letter 'z' followed by the letter 'a') in the alphabet. For example if $k = 4$, then the letter 'a' in plaintext becomes 'e' in ciphertext, 'b' in plaintext becomes 'f' in ciphertext and so on. The plaintext 'computer' will become a ciphertext 'gtqryxiv'. Here, the value of k serves as the key. In symmetric key encryption, divulging the key means anyone can encrypt and decrypt messages. We must therefore keep the key secret.

The following is a scenario using symmetric encryption:

1 Alice and Bob agree to use the Caesar cipher to communicate, and pick k as the secret key.

2 Bob uses the Caesar cipher to encrypt a confidential message to Alice.

3 Bob sends the encrypted message to Alice.

4 When she receives Bob's mail, Alice decrypts the message and reads the confidential message.

The above scenario is depicted in the following figure.

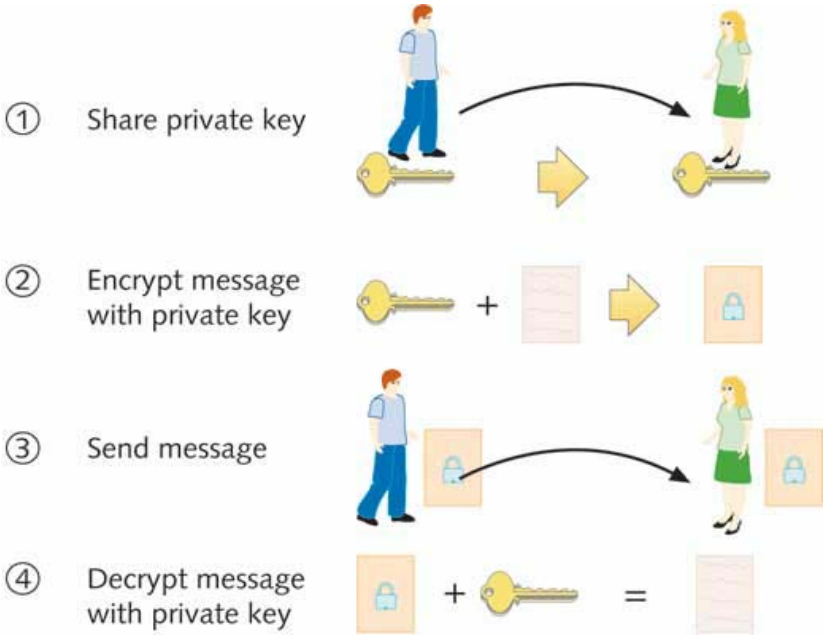


Figure 8.9 Private key encryption (Dean 2004, Figure 14-6)

Symmetric key systems are much faster than asymmetric systems, but they have the following two major drawbacks:

- *Key distribution* — A secure channel is required by which the correspondents can agree on a key before their first encrypted communication.
- *Scalability* — It is difficult to manage the secret keys because of the growth in the number of secret pairs.

Some examples of symmetric key algorithm are:

- Advanced Encryption Standard (AES), a recommendation of the National Institute of standards and Technology (NIST) with the key length at 128 bits or above
- Triple Data Encryption Standard (DES), a recommendation of NIST in the past
- Blowfish
- RC4, RC5 and RC6.

Self-test 8.4

1 What is the ciphertext of the following message using Caesar cipher with $k = 3$?

```
'The port # is 21'
```

(*Hint*: You can reference the ASCII codes table at http://en.wikipedia.org/wiki/ASCII#ASCII_printable_characters. After shifting a space character for 3 places (k = 3), it becomes '#'.)

2 What is the plaintext message of the following ciphertext message with $k = 3$?

'L#oryh#brx#Dolfh/#Ere'

Activity 8.4

In the UNIX system, the encryption command `crypt` performs the encryption processes. `Crypt` reads from the standard input and writes on the output. The password is a key that selects a particular transformation. If no password is given, `crypt` demands a key from the terminal and turns off printing while the key is being typed in. `crypt` encrypts and decrypts with the same key.

You can try to encrypt and decrypt a file by using the command `crypt`.

1 Edit a file to, say `secret.txt`. Then you can encrypt the `secret.txt` such as:

```
$ crypt key < secret.txt > encrypted.file
```

2 Display the content of this encrypted file.

3 After that, you can send the encrypted file to your receiver. It is relatively safe from anyone peeking at the content of the file.

4 At the receiving end, the receiver performs the decryption step to retrieve the file.

```
$ crypt key < encrypted.file > scret.txt
```

5 Display the content and compare it with the original `secret.txt` file.

Asymmetric encryption

In an asymmetric key encryption system, which is also known as a **public key** system, a pair of keys is used for encryption and decryption. If a message is encrypted by one key, we have to use the other key to decrypt the message. It is *not* possible to use the same key for encryption and decryption in an asymmetric key system. In a public key system, the pair of keys is made up of one public key and one private key. The public key is made known to everyone, while the **private key** must only be known to the owner.

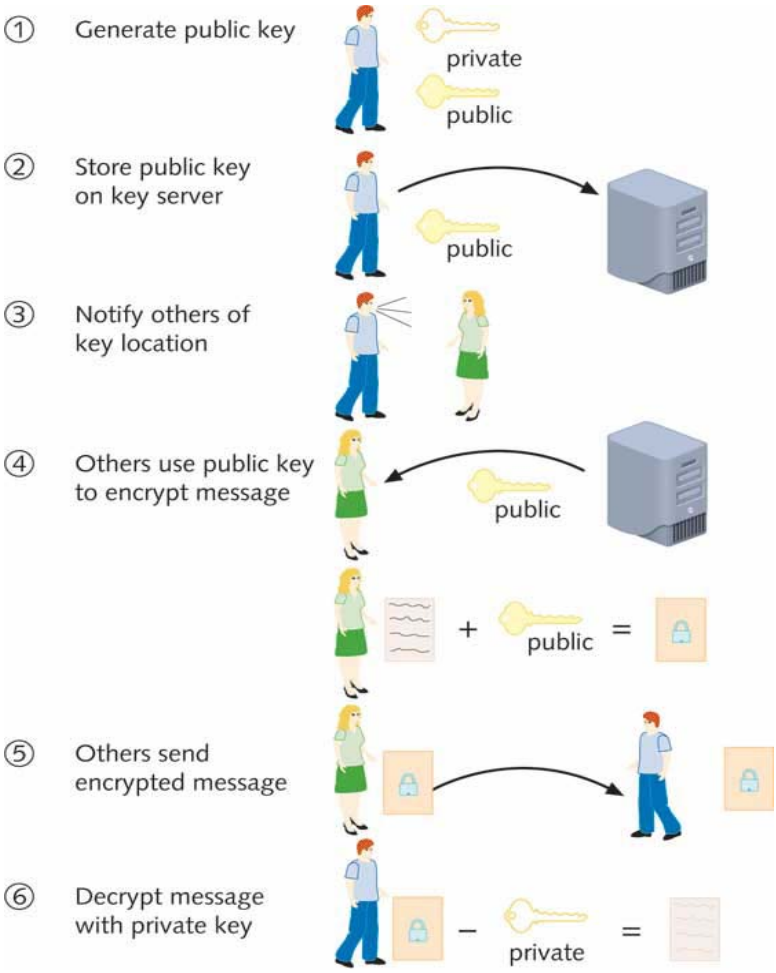


Figure 8.10 Public key encryption

The public and private keys are mathematically related in such a way that the ciphertext encrypted by one key can only be decrypted by the other key. Also, it is very difficult to derive a key from one another. That means even if you have Bob's public key in hand, you have no way to deduce Bob's private key, which is known to Bob only.

The following outlines the strengths and weakness of asymmetric key encryption:

Strengths:

- easier key distribution
- fetter scalability
- can provide confidentiality, authentication and **non-repudiation**.

Weaknesses:

- works much slower than symmetric systems.

Some examples of symmetric key algorithm are RSA, Diffie-Hellman, Elliptic Curve Cryptosystem (ECC).

In the sections that follow, we explain how to use public key encryption to achieve confidentiality, authenticity and integrity.

Confidentiality

Let's say Bob uses Alice's public key to encrypt a message and send the encrypted message to Alice. When Alice receives the message, she uses her private key to decrypt the message and read its content. If the encrypted message is intercepted, its content will not be divulged since the interceptor doesn't have Alice's private key. This achieves confidentiality.

Authenticity and non-repudiation

Bobs uses his private key to encrypt a message and send the encrypted message to Alice. When Alice receives the message, she uses Bob's public key to decrypt the message and read its content. Alice is assured that the message must be issued by Bob, and Bob can't repudiate the issuance since the message can be decrypted by Bob's public key. This achieves authenticity and non-repudiation.

In practice, however, we seldom encrypt a whole message since it is very time-consuming. Rather, digital signing technique will be used to ensure authenticity and non-repudiation, which is explained below.

Integrity

A **hash function** is a one-way mathematical function that can take a variable-length message as input and convert it into a fixed-length (generally smaller) hash value output, known as a **message digest**. A hash function holds such a one-way characteristic that computing a message digest is easy, but reversing the message digest back to its originating message is very hard. Another key characteristic is that if there is any change in the input message, the hash function will compute a different hash value. In combination, these characteristics make the

digest the fingerprint of the message. If two messages can produce the same digest, then their contents should be identical.

Following the above examples, let's say Bob produces a digest of a document using a hash function. He then encrypts the hash with his private key, which results in a digital signature. He sends the document and the signature (i.e. the encrypted message digest) to Alice over the network. On the receiving side, Alice computes a message digest from the received document and decrypts the digital signature with Bob's public key to retrieve the message digest of the original document. If the two message digests (one from computation, one from retrieval) are the same, the received document should be tamper free, and it must have been issued by Bob (since decryption is done with Bob's public key). If they are different, however, it means the received document was tampered with while in transit.

The digital signature helps achieve integrity, authenticity and non-repudiation.

The RSA algorithm

RSA, named after its inventors Rivest, Shamir and Adleman, was the first full-fledged public key algorithm that could work for encryption and digital signing. It is not difficult to understand and implement, while its protection remains robust in today's standards.

RSA gets its security from the difficulty of factoring large numbers, especially those composited of prime number factors. The public and private keys are functions of a pair of large (i.e. they could be over a hundred binary digits) prime numbers. Recovering the plaintext from the public key and the ciphertext is equivalent to factoring the product of the two primes, which has been proved to be computationally difficult.

In this context, let's say Bob chooses the public key as a pair of numbers (N, e), while the private key is a pair of numbers (N, d) such that $N = p \times q$ where p and q are prime numbers. The numbers e and d are mathematically related to N, but such a relationship is not too important for a basic understanding of the algorithm. Bob can let anyone know his public key (N, e) but he has to keep his private key (N, d) secret.

The protocol goes on as follows:

1 Alice uses Bob's public key (N, e) to encrypt the plaintext P:

$$C = P^e \bmod N$$

Note: 'x mod n' means the remainder of x when divided by n. '23 mod 10' is 3. mod is also known as modulo.

2 Alice sends the ciphertext C to Bob.

3 Bob uses his private key to decrypt the ciphertext:

$$P = C^d \bmod N$$

A short example will probably go a long way to explain the above.

If p = 47 and q = 71, N = pq = 3337. The encryption key (N, e) and decryption key (N, d) are chosen (with some calculations done behind) as (3337, 79) and (3337, 1019).

To encrypt the message 'FHH', then P = 688 (F is 6th letter), C = $688^{79} \bmod 3337 = 1570$

To decrypt the message, $C^d \bmod N = 1570^{1019} \bmod 3337 = 688$ (i.e. P)

You can use the calculator application (select 'scientific mode') included with the Windows desktop to verify the above calculations. It supports the operation of raising x to the power of y and modulo calculation, as indicated in the figure below.

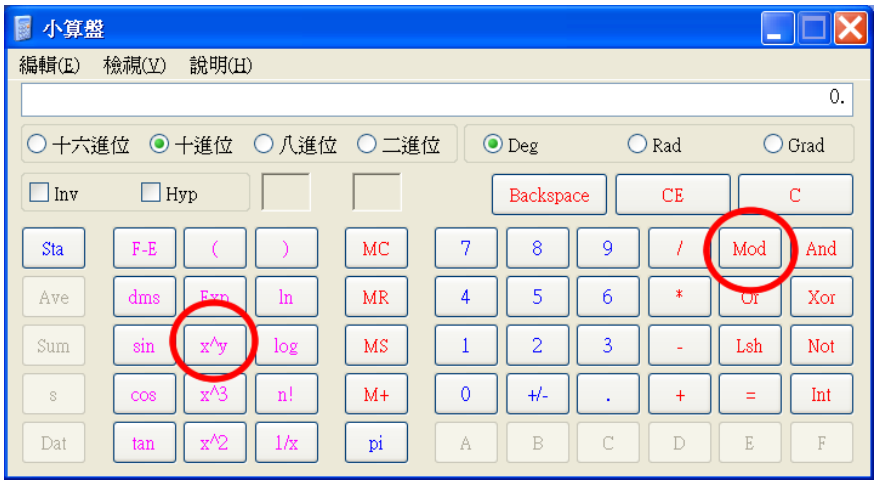


Figure 8.11 The useful scientific calculator provided in Windows XP

The above example shows RSA's capability to perform encryption using Bob's public key to address confidentiality.

The American Bar Association has published a tutorial related to digital signatures. Digital signature technology may greatly affect legal issues; perhaps digital signatures will be accepted as a legally recognized form. This document also introduces the concept of digital signature technology. Attempt the tutorial in the following reading.

Reading 8.4 (online)

American Bar Association, *Digital Signature Guidelines Tutorial*:

http://www.abanet.org/scitech/ec/isc/dsg-tutorial.html

Self-test 8.5

1 What are the strengths of asymmetric key algorithm? And what is its major weakness?

2 Provide some examples of asymmetric key algorithms.

3 Using the RSA algorithm, encrypt the message 'BE' with key pair (15, 3) and (15, 5).

Hybrid encryption approach

In practice, we often use asymmetric and symmetric key algorithms in a complementary manner to cover up each other's shortcomings. A symmetric algorithm creates secret keys that are used for encrypting bulk data (since symmetric algorithms are faster), while an asymmetric algorithm is used for distributing the secret keys to target participants. This is illustrated in the figure below.

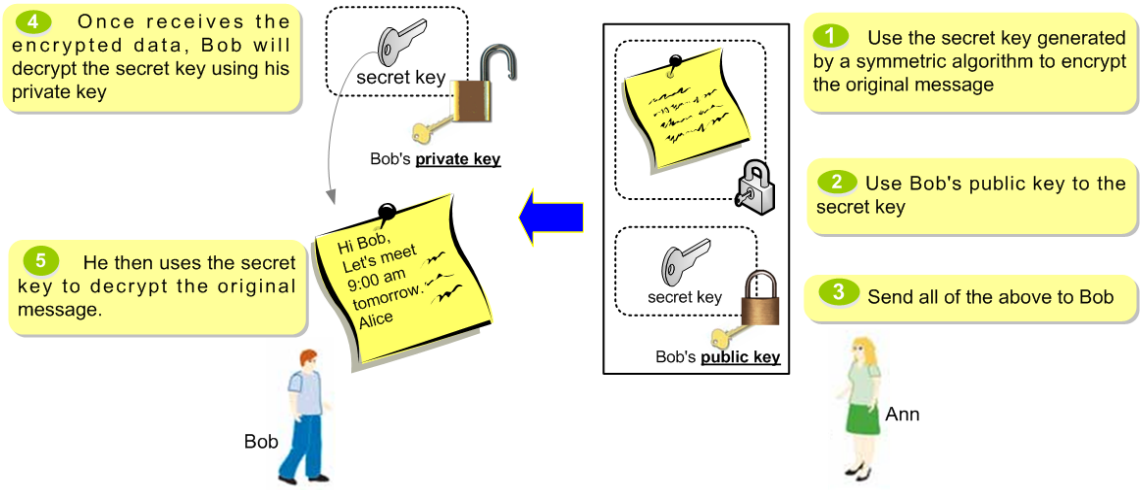


Figure 8.12 Hybrid approach using both asymmetric and symmetric key algorithms

Again, be sure you read the numbered text in sequence to follow the scenario in which Ann sends a message to Bob using the hybrid encryption approach.

When a secret is used for only one communication session between users, it is also called a session key. A new session key should be generated when a new communication session is started, and it will be distributed to involved parties in a manner similar to the above scenario. Frequent changes of session keys can lower the chance of encrypted communications being compromised.

The following reading describes the process of key encryption. Both private and public key encryptions are discussed. In private key

encryption, data are encrypted using a single key that only the sender and the receiver know. In public key encryption, data are encrypted using two keys: one is a key known only to a user (private key), and the other is a public key associated with the user.

Reading

Dean (2010) 595–600.

Self-test 8.6

Compare and contrast private key encryption and public key encryption on such attributes as keys, key exchange, speed, use and security services provided.

Public key infrastructure (PKI)

When private and public keys grow to a sizeable number, it is not an easy task to manage them. You may then have doubts about whether Bob's public key in your hand is still valid (i.e. that it has not expired), or you may run into the problem of mixing Bob's public key with Alice's.

To solve these problems, the public key infrastructure (PKI) should be established. PKI is made up of a number of consistent components including digital certificates (i.e. X.509), certificate authorities, keys, users, governments, cryptography technologies, etc. A **digital certificate** is a password-protected and encrypted file that holds an individual's identification information, including a public key. The certificate is digitally signed and issued by a trusted third party, i.e. the **certificate authority (CA)**. The CA is responsible for verifying the identity of the key owner and for distributing the owner's digital certificate for use by others. Alice can therefore obtain a copy of Bob digital certificate from the issuing CA, and retrieve Bob's public key from the certificate and use it to encrypt messages for Bob or to authenticate Bob's messages to her.

The Hong Kong government laid a foundation for the deployment of PKI through the enactment of the Electronic Transactions Ordinance (ETO) in early 2000 and through the establishment of a public Certification Authority (CA) through the Hongkong Post. Under the ETO, a digital signature created by a digital certificate issued by a recognized CA, including for example Hongkong Post, carries the same legal status as its hand-written equivalent. Buyers and sellers must execute a transaction authorized by a digital signature, and they cannot reject or repudiate a digital signature on the sole ground of such signature being in an electronic format.

The following reading introduces the concept of digital certificates. It explains how to obtain a digital certificate from a certification authority.

Reading 8.5 (online)

VeriSign Australia, 'Introduction to digital certificates':

https://www.comodo.com/resources/small-business/digital-certificates-intro.php

The reading below describes how to use the e-Cert, which is the digital certificate issued by the Hong Kong Post Office, who is one of the local CAs. e-Cert can help secure email communications, online government services, banking services, stock trading and other transactions.

Reading 8.6 (online)

Hong Kong Post, 'Usage of e-Cert — Wide usage in your everyday life':

http://www.hongkongpost.gov.hk/product/ecert/usage/index.html

Activity 8.5

Complete Step 1 of 'Lab 1.4 — Networking III: Connecting to Unix Servers with SSH Public Key Authentication' from your textbook *A Practical Approach to Internet Programming and Multimedia Technologies*. This lab gives you hands-on experience in using public key technologies for authenticating yourself to servers. This authentication method is an alternative to using passwords, and offers a more secure protection.

Network security protocols

In this section we'll cover some network security protocols designed to protect data while they are in transit to ensure they remain unchanged and private throughout the transmission. These protocols are often used on the Internet.

SSL

The **Secure Socket Layer (SSL)** is a protocol developed by Netscape to provide a security sub-layer between application protocols (such as FTP, HTTP, or Telnet) and TCP/IP; it is shown in Figure 8.13 below.

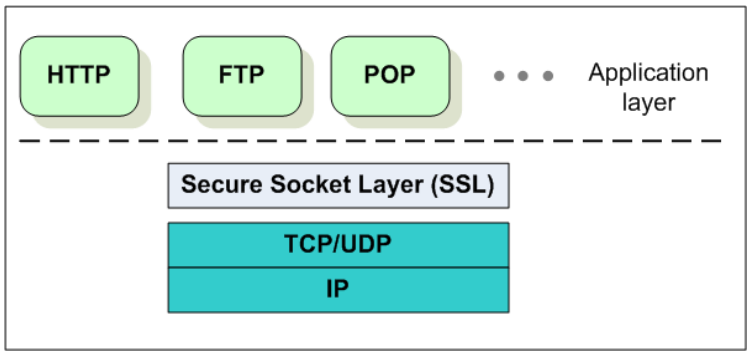


Figure 8.13 Secure Socket Layer (SSL) protocol

SSL provides data encryption (commonly using 128-bit or 256-bit data encryption) and authentication between servers and clients. SSL forms a secure connection between a server and a host, and all data transmitted over the connection will be encrypted. Common browsers including Internet Explorer, Firefox, etc. support SSL. SSL is widely used to support on-line transactions to allow customers to place orders over the Internet securely. A URL that starts with https:// indicates that it is a secure connection.

HTTPS (HTTP over Secure Sockets Layer) uses TCP port 443, rather than port 80, which is used by the HTTP protocol. The SSL protocol uses a digital certificate to authenticate one end or both ends of transactions.

> ***Reading 8.7***
>
> *i-Net Guide+*, 3rd edn, 655–56.

SSH

Secure Shell (SSH) provides secure remote connections to hosts. It allows users to log on to a host, execute commands on that host, and copy files to or from that host. In the past, we used telnet to do such tasks.

However, telnet provides little protection for transmitted data, including passwords. SSH circumvents this shortcoming by encrypting all data to counteract the risk of data interception. In addition, SSH performs authentication on the clients and hosts to guard against attacks set up by forged identities.

SSH must first generate public and private keys on the client's workstation, and carry out the key exchange with the host. Subsequently, these keys can be used for authentication and data encryption. All modern Unix and Linux systems come with the SSH suite of protocols. Windows users can download some free SSH clients such as PuTTy to connect to SSH-enabled hosts.

SSH is highly configurable and supports port-forwarding function, which allows users to exchange application traffics (e.g. http, ftp) over an SSH-secured port for higher security. SSH uses TCP port 22 for communication.

SCP and SFTP

The Secure Copy Protocol (SCP) is an extension to SSH to allow you to copy files from one host to another securely. Like the relationship between SSH and telnet, SCP aims to alleviate the unsecure operation of the File Transfer Protocol (FTP),which transmits file contents, usernames, and password in plaintext over networks. The SFTP (Secure File Transfer Protocol) is used by some proprietary SSH implementations. SFTP is slightly different from SCP as it can not only copy files, but also list files, change directories, etc. To use SFTP on a Unix host to access files located in another host, simply type 'sftp < target-host-name>'. SFTP encrypts all transmitted data.

WinSCP is the freeware client that allows Windows XP to transfer files to or from a host using SCP or SFTP services. You will try it out in the upcoming Activity 8.6.

Reading

Dean (2012) 521–23.

Self-test 8.7

1 Briefly describe the process through which a browser (client) initiates a secure SSL connection with a server.

2 In the SSL protocol, which of the symmetric and asymmetric algorithms is used for encrypting the bulk of data for transmission? And which is used for key distribution?

Activity 8.6

To try out the SCP/SFTP protocol, you can download a copy of WinSCP from http://winscp.net/eng/download.php and install it onto your Windows desktop. Configure it to access your files and directories located in the ucourse2.ouhk.edu.hk server as follows:

Host name :	ucourse2.ouhk.edu.hk
Port number:	22 (default)
User name:	Your account id for the ucourse2 server
Password:	Your account's password
Private key file:	Leave it blank

Press **Login** to start the login process.

Using WinSCP is easy since it uses a graphical interface. You can use this utility to access any server that supports SCP or SFTP (e.g. labsupport.no-ip.org server). If you have any problem with this activity, you can contact your group's tutor for assistance.

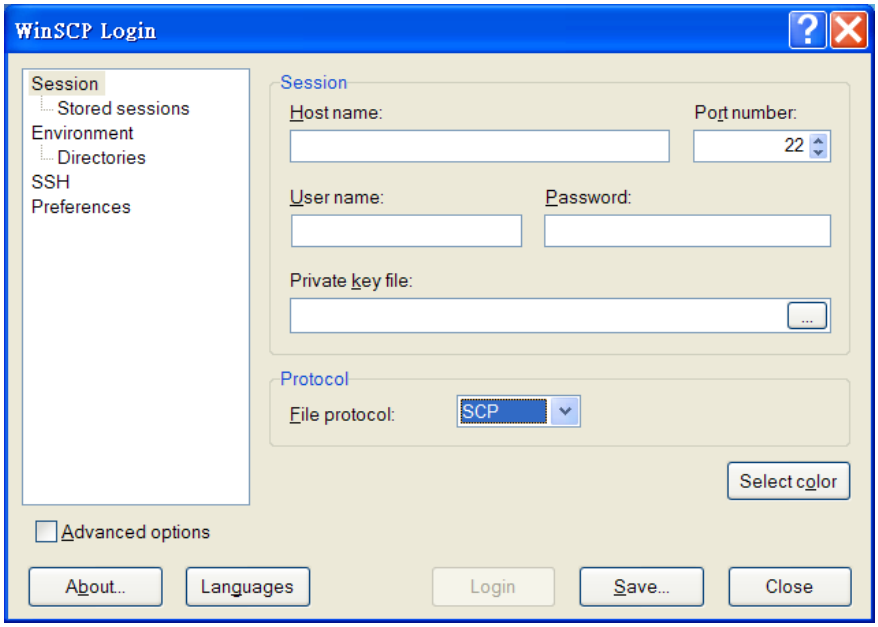


Figure 8.14 The configuration of WinSCP utility

Authentication protocols

Authentication protocols are rules that computers follow to accomplish authentication in order to ascertain a user's identity before granting him the requested access to servers and other resources. The most common authentication protocols are described below:

RADIUS

Remote Authentication Dial-In User Service (RADIUS) provides centralized network authentication and accounting for multiple users. A RADIUS server does not replace the functions provided by the remote access server, but communicates with the access server to manage user logons. RADIUS is often used with dial-up networking connections, and it provides encryption to prevent users' IDs and passwords from being divulged. The figure below shows how a RADIUS operates in a network environment.

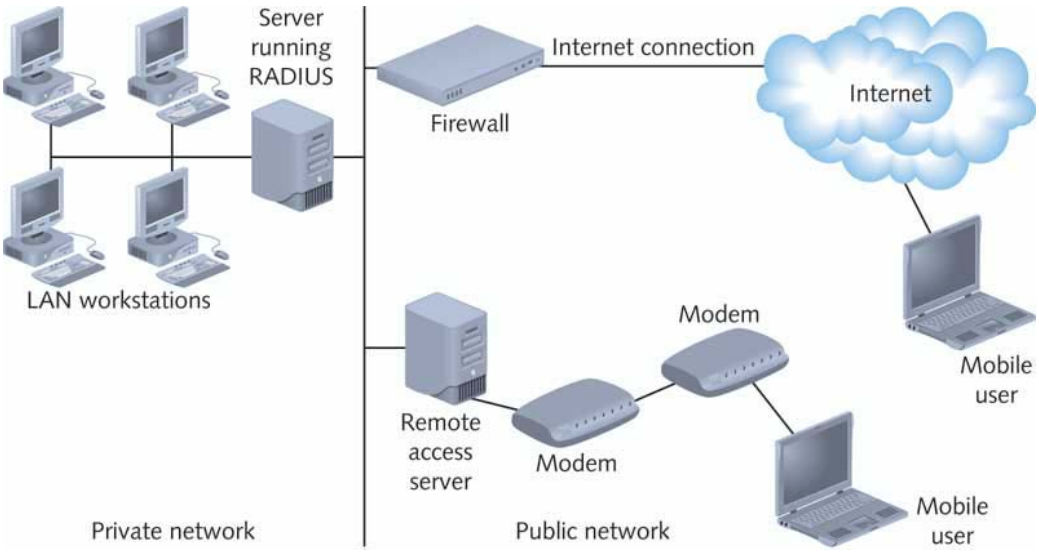


Figure 8.15 A RADIUS server providing centralized authentication (Dean 2004, Figure 14-8)

PAP, CHAP and MS-CHAP

As alternatives to using centralized authentication protocols, there are other authentication protocols that individual servers can use to authenticate their users.

Password Authentication Protocol (PAP) is an authentication protocols for use over a point to point link. Upon establishing a point-to-point link with a client, the server will use PAP to verify the credentials (e.g. username, password). It then grants access accordingly. PAP is not secure, however, since it passes the credentials in clear text.

Challenge Handshake Authentication Protocol (CHAP) is a more secure authentication protocol since it encrypts passwords before transmitting them over networks. It involves a three-way handshake procedure, as shown in Figure 8.16, for authentication in which a randomly generated string, called the challenge, is used to encrypt the password.

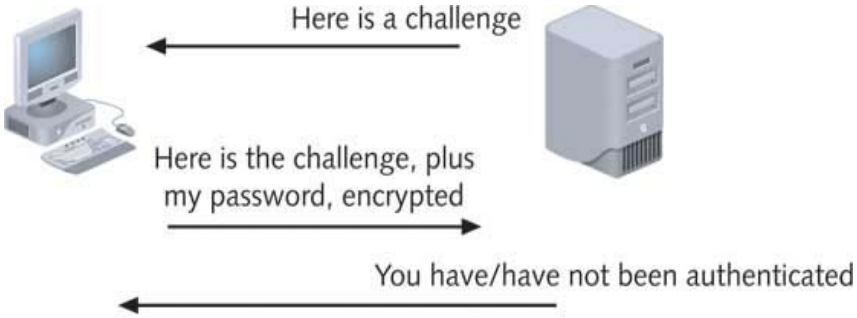


Figure 8.16 Three-way handshake used in CHAP (Dean 2004, Figure 14-10)

Microsoft Challenge Authentication Protocol (MS-CHAP) is a similar authentication that is for use in Microsoft-based systems only. Microsoft Challenge Authentication Protocol, version 2 (MS-CHAPv2) is more secure than MS-CHAP in that it requires both the server and client to authenticate each other, instead of only the server authenticating the client, as other previously mentioned protocols do.

Kerberos

Kerberos is a computer network authentication protocol developed at the Massachusetts Institute of Technology (MIT) in early 1980s that allows individuals communicating over a network to prove their identities to one another in a secure manner. Kerberos provides mutual authentication — both the user and the service verify each other's identities. Firewalls can counteract the risks arising from the external environment, while Kerberos provides the means to protect network services against unauthorized uses that may arise from the internal environment. Yes, the internal environment — never think that the internal environment is always safe and free of risk! Just read the following news from 2008 about a contractor that tried to exceed its authorized access to steal some sensitive data and sell them for profit.

A case of internal security incidents

Network World (2008) describes a real case of internal data theft that happened to the US military. It provides evidence of why we can't assume the internal environment will always be secure, and why we have to implement authentication (e.g. Kerberos) or other internal protection measures.

> What better way to thank members of the United States military for their patriotic service than by stealing their identities and selling them to undercover government agents? Randall Craig, a 41-year-old

Houston man who worked as a private contractor at a San Antonio Marine Reserve Corps Center, pleaded guilty in May 2008 to exceeding authorized access to a computer and aggravated identity theft after he sold the names and Social Security numbers of 17,000 military employees to an undercover FBI agent. Earlier in that year, Craig had met with the agent at the Houston airport to discuss selling him a thumb drive that contained the employees' data on it. Eventually he agreed to sell the information for US$500.

The Kerberos protocol is built primarily on symmetric key cryptography. All credentials and authentication information are properly encrypted before passing over the network between the client and the services. Modern NOSs adopt Kerberos with the purpose of allowing a valid user to use the network services that she is entitled to use throughout her logon session. For instance, Windows Servers 2000, 2003 and 2008 use Kerberos as their default authentication method, while Red Hat Enterprise Linux 4 and later versions use Kerberos in both client and server versions.

We'll now discuss the basic operation of Kerberos. You can treat it as an optional reference.

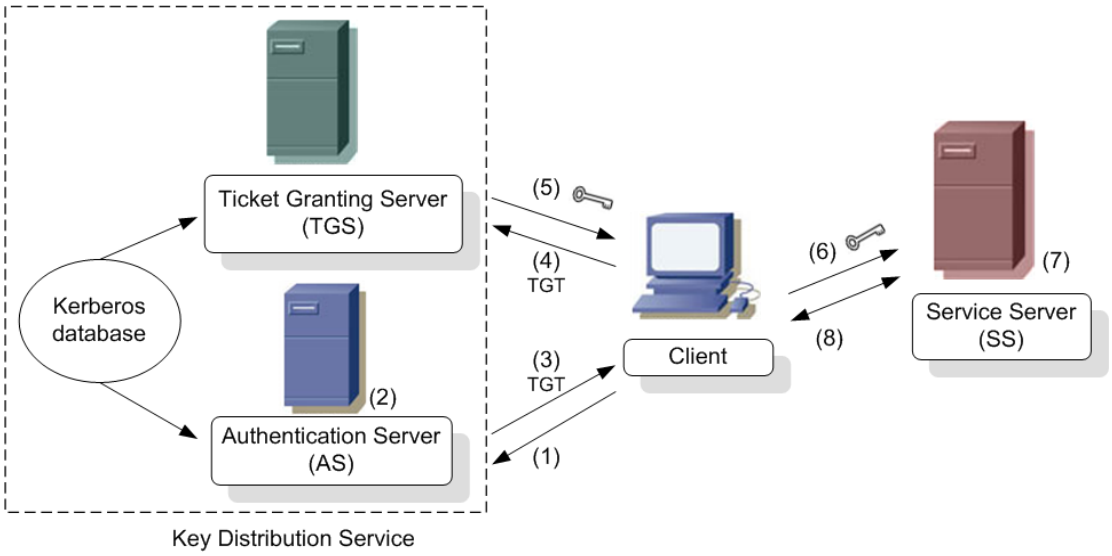


Figure 8.17 The Kerberos system

The following steps explain the Kerberos system illustrated in Figure 8.17:

1 The Authentication Server (AS) receives a request by the client and verifies that the client is what it claims to be, which is usually done by a database lookup of the user's ID.

2 Upon verification, a *timestamp* is created. This puts the current time in a user session, along with an expiration date. The default expiration date of a timestamp is eight hours. The encryption key is then created. The timestamp ensures that when eight hours is up, the encryption key is useless. This is for better security.

3 The key is sent back to the client in the form of a *Ticket-Granting Ticket* (TGT). This is a ticket that is issued by the AS to be used for authenticating the client for future reference.

4 The client submits the TGT to the Ticket Granting Server (TGS) to get authenticated.

5 The TGS creates an encrypted client/server session key with a timestamp. The TGS sends the encrypted session key and a service ticket which contains another copy of the session key but which has been encrypted by the Service Server (SS)'s private key to the client.

6 The client decrypts the client/server session key for future use while it also sends the service ticket to the SS.

7 The SS decrypts the service ticket to get the client/server session key, and makes sure the timestamp of the ticket is still valid. If it is, the SS sends back a confirmation message encrypted with the session key to the client.

8 The client decrypts the message with the session key and checks if the timestamp is updated correctly. If so, communication is initiated between client and server.

Reading

Dean (2012) 524–31.

Self-test 8.8

1 What are the key benefits of Kerberos?

2 Name two operating systems that support Kerberos.

Summary

In this unit you investigated a number of fundamental concepts and issues surrounding computer security. Among all of the many topics that computer security is concerned with, we kept our focus of attention on network security.

To begin with, you reviewed some high level topics including fundamental security goals and security management practice. This provided you with perspective on how to deal with the topics that followed, and to relate them with one another.

What followed was our exploration of a host of techniques and technologies that can make a network design more secure. They include network segmentation, fault tolerance, firewalls, anti-virus tools, IDS, VPN, etc. These tools should be built into a network at the design stage, instead of being installed to a network as after-thoughts once the implementation is complete.

You went on to study the security in NOSs, which is important since any flaw in the NOS will simply transpose into a threat to all applications running on top of it. Just recall the saga caused by the worm Red Code in early 2000s that plagued hundreds of thousands of Windows-based Web servers around the globe!

The next topic was cryptography, which is the cornerstone in modern computer security. Many security protocols or services made use of cryptography to promote confidentiality, integrity and authenticity — the fundamental goals of security. We discussed symmetric and asymmetric encryptions and looked into a few cryptographic algorithms including RSA — the first full-function public key encryption algorithm. There was also a lab to practice the public key authentication.

In practice, we often have to pick and choose among appropriate security protocols and make them work for us. Secure Socket Layer (SSL), Secure Shell (SSH) and Secure Copy Protocol (SCP) are among the most popular security protocols in use nowadays. We discussed each of them. The unit ended with a discussion of some authentication protocols such as Kerberos that are used by Windows servers, Linux and Unix.

This has been a fairly packed unit indeed, covering a wide range of topics. After working through it, you should now have a clearer picture of network security and be prepared to take a proper approach to deal with it.

Suggested answers to self-tests and activities

Self-test 8.1

1 The four key security goals are confidentiality, integrity, availability and authenticity.

2

Risk descriptions	Rankings	Suggested countermeasures
Leakage of customers' personal information to the Internet via employees' home PCs	High priority	*(Reduction)* Include the prohibition of home PCs for business purposes in the organization's security policy. *(Reduction)* Provide notebooks with proper security protection to employees who need to work at home.
Network intrusion by hackers	High priority	*(Reduction)* Implement firewalls and intrusion detection systems.
. . .	. . .	. . .
Flood damage	Medium priority	*(Assignment)* Take out insurance.
File server running out of hard disk space	Medium priority	*(Reduction)* Issue capacity planning guidelines.
. . .	. . .	. . .
System disruption caused by earthquakes	Low priority	*(Acceptance)* No action.
. . .	. . .	. . .

3 A security policy should not pertain exclusively to computers or networks. For example, it might include a human resources requirement on security training so that every computer user would have a reasonable level of awareness and understanding on the subject of computer security. This can help to increase the acceptance of the policy among employees. Also, it might state a physical security requirement such as requiring employees to wear a valid staff badge in order to gain access to any office premises.

Self-test 8.2

1 The modified firewall rules table at Denver should be:

```
Denver firewall rules
#1 ALLOW
Direction:   IN
Source IP: 202.198.6.1, Source port: any
Dest. IP:    202.198.6.2, Destination port: 1723
#2 ALLOW
Direction:  OUT
Source IP: 202.198.6.2, Source port: any
Dest IP:     202.198.6.1, Destination port: 1723
#3 ALLOW
Direction:   IN
Source IP: 202.198.6.3, Source port: any
Dest. IP:    202.198.6.2, Destination port: 1723
#4 ALLOW
Direction:  OUT
Source IP: 202.198.6.2, Source port: any
Dest IP:     202.198.6.3, Destination port: 1723
#5 DENY
All other traffics
```

2 Proxy servers operate at the application layer.

3 The major advantage of VPNs over other alternatives such as dedicated point-to-point connection (e.g. T1 links) is their relatively lower cost, thus making VPN an attractive and practical option to businesses of different sizes.

4 The two operation modes of IPSec are transport mode and tunnel model.

Self-test 8.3

Full backup — All data files on a server are stored to backup media with their ‘archive’ bit unchecked. This type of backup takes the longest time to complete.

Incremental backup — All data files which have been changed since last full or incremental backup are stored to backup media with their ‘archive’ bit unchecked.

Differential backup — All data files which have been changed since last full backup are stored to backup media *but* with their ‘archive’ bit remaining unchanged.

A sound backup strategy will usually execute full backups at regular intervals (e.g. every week) while it will also intersperse some incremental or differential backups in each interval (e.g. every day). This combination can lessen the frequency of full backups for better efficiency, while maintaining the integrity of backup protection.

Self-test 8.4

1 Wkh#sruw#&#lv 54

2 I love you Alice, Bob

Self-test 8.5

1 Strengths:

- better key distribution than symmetric systems
- better scalability than symmetric systems
- san provide confidentiality, authentication and nonrepudiation.

Major weakness:

- works much slower than symmetric systems.

2 Some examples are RSA, Elliptic Curve Cryptosystem (ECC) Diffie-Hellman.

3 To encrypt the message 'BE', i.e. P = '25' (B is the 2nd letter, and E the 5th), (N,e) = (119, 5), (N, d) = (119, 77)

The answer is: $C = 25^5 \bmod 119 = \underline{9}$

Verifying the answer: $P = 9^{77} \bmod 119 = 25$

Self-test 8.6

Attributes	**Symmetric**	**Asymmetric**
Keys	One key is shared between the sender and receiver	Two keys are used
Key exchange	Out-of-band method	Public key can be exchanged openly
Speed	Faster	Slower
Use	Encrypt bulk of data	Distribute keys and sign documents
Security service provided	Confidentiality	Confidentiality, authentication, non-repudiation and integrity

Self-test 8.7

1 The process is as follows.

   a A browser requests a secure page (usually https://).

   b The Web server sends its public key with its certificate.

   c The browser checks that the certificate was issued by a trusted party (usually a trusted root CA), that the certificate is still valid, and that the certificate is related to the site contacted.

   d The browser then uses the public key to encrypt a random symmetric encryption key and sends it to the server with the encrypted URL required, as well as other encrypted http data.

   e The Web server decrypts the symmetric encryption key using its private key and uses the symmetric key to decrypt the URL and http data.

   f The Web server sends back the requested html document and http data encrypted with the symmetric key.

   g The browser decrypts the http data and html document using the symmetric key and displays the information.

2 In SSL, the symmetric algorithm is used for bulk encryption, while the asymmetric algorithm is for key distribution.

Self-test 8.8

1 The following are the key benefits of Kerberos:

   a Avoiding password sniffing — Kerberos will prevent plaintext passwords from being transmitted over the network.

   b Easing the maintenance and use of large account databases — Kerberos will centralize username and password information, which will make it easier to maintain and manage this data. Different applications can make use of such data rather than each of them keeping their own set of data redundantly.

   c Protecting password filename/database from theft — Kerberos avoids the need for storing password information locally on a machine, whether it is a workstation or server, thereby reducing the likelihood that a single machine compromise will result in additional compromises. For instance, when a user requests for a service from a server, he will present a valid ‘ticket’ obtained from the KDS (Key Distribution Service) to the server without divulging his password. Therefore the server does not keep any user’s password.

2 The following operating systems support Kerberos:

- Windows Server 2000 and later versions;
- Red Hat Enterprise Linux 4 and later versions; and
- Sun's Solaris.

Activity 8.1

1 The contents of the log file should look like the following:

```
#Version: 1.5
#Software: Microsoft Windows Firewall
#Time Format: Local
#Fields: date time action protocol src-ip dst-ip src-port dst-port size tcpflags tcpsyn
tcpack tcpwin icmptype icmpcode info path

2009-01-09 15:22:12 DROP TCP 216.239.122.178 192.168.94.224 443 1157 40 R 0 0 16384 - -
- RECEIVE
2009-01-09 15:22:43 DROP TCP 219.77.10.29 192.168.94.224 8080 4717 40 R 3712237155 0
17171 - - - RECEIVE
2009-01-09 15:23:40 OPEN TCP 192.168.94.224 216.239.122.178 1158 80 - - - - - - - - -
2009-01-09 15:23:44 OPEN UDP 192.168.94.224 210.0.128.250 53608 53 - - - - - - - - -
2009-01-09 15:23:44 OPEN TCP 192.168.94.224 209.85.143.104 1159 80 - - - - - - - - -
2009-01-09 15:23:44 OPEN UDP 192.168.94.224 210.0.128.250 52652 53 - - - - - - - - -
2009-01-09 15:23:44 OPEN TCP 192.168.94.224 64.233.189.104 1160 80 - - - - - - - - -
2009-01-09 15:23:53 CLOSE TCP 192.168.94.224 216.239.122.178 1156 80 - - - - - - - - -
2009-01-09 15:24:08 CLOSE TCP 192.168.94.224 64.233.189.104 1160 80 - - - - - - - - -
2009-01-09 15:24:08 CLOSE TCP 192.168.94.224 64.233.189.99 1161 80 - - - - - - - - -
2009-01-09 15:24:08 CLOSE TCP 192.168.94.224 209.85.143.104 1159 80 - - - - - - - - -
2009-01-09 15:24:08 CLOSE TCP 192.168.94.224 209.85.173.113 1162 80 - - - - - - - - -
```

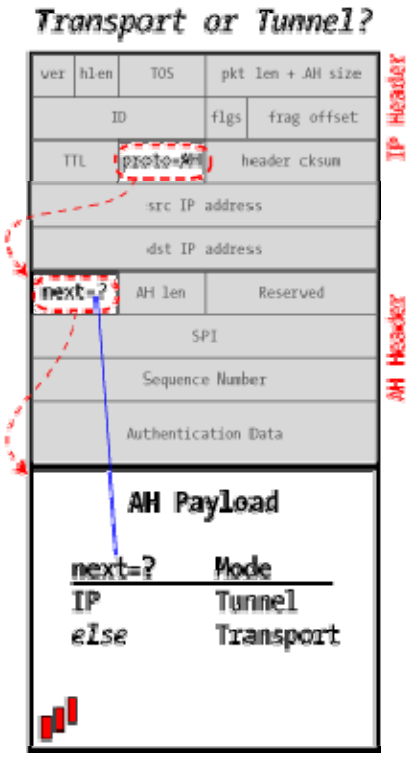
Observations:

- For the first two packets, the firewall *drops* their associated connections since they may come from malicious external hosts.
- For the next five packets, the firewall opens the connections for them in order to access some Web servers through port 80, and some Domain Names Server (DNS) through port 53, respectively.
- For the remaining packets, the firewall closes the connections for them since they are entailed in the TCP (connection-oriented) communication mode. There is no close connection for UDP packet. Do you know why?

Activity 8.2

1 What distinguishes Transport mode from Tunnel mode is the next header field in the AH header. When the next-header value is IP, it means that this packet encapsulates an entire IP datagram (including the independent source and destination IP addresses that allow separate routing after de-encapsulation). This is Tunnel mode. Any other value (TCP, UDP, ICMP, etc.) means that it's in Transport mode and is securing an endpoint-to-endpoint connection. The top-level of the IP datagram is structured the same way regardless of

mode, and intermediate routers treat all flavors IPsec/AH traffic identically without deeper inspection.



3 If the Web browser has configured with the use of proxy server, you can check it from the browser configuration. Assuming the browser is Internet Explorer, you should follow the steps listed below:

a Click on **Tools** (工具). Select **Internet Options** (網際網絡選項)

b Click **Connectivity** (連線) and then click **LAN Settings** (區域網絡設定).

c At **Proxy** (伺服器), click **View**, which is to the right of the manual proxy button.

d You will find the corresponding proxy server names.

If the browser has not been configured to use the proxy server, you may need to ask the network administrator for the proxy server information.

Glossary

availability — Ensuring that information is ready for use as and when required by users.

authentication — The process of validating the claimed identity of an end-user or a device such as a host, server, switch, router, and so on.

authenticity — Ensuring that data access, transactions and exchanges are verified and made accountable.

CERT — Computer Emergency Response Team, a formal organization of system administrators whose members provide services pertaining to issues related to computer and network security.

certificate authority (CA) —An entity trusted to sign digital certificates and, therefore, to vouch for the identity of others.

cipher — A procedure that transforms data between readable text and ciphertext; a cryptographic algorithm.

ciphertext — Encrypted text that must first be decrypted to produce readable text.

confidentiality — Ensuring that information is protected against disclosure to people who are not explicitly intended to receive it.

cryptography — The science of writing or reading coded messages.

demilitarized zone (DMZ) —A dedicated physical or logical network that exposes an organization's external services (e.g. Web) to the Internet.

digital certificate — A password-protected and encrypted file issued by a CA that holds an individual’s identification information, including a public key.

encryption — A method of scrambling information in such a way that it is not readable by anyone except the intended recipient, who must decrypt it to read it.

firewall — A specialized device or a software program that selectively filter traffics passing between networks in order to block out any attempt to attack. It operates mainly at the network layer.

hash function — A mathematical computation that results in a fixed-length string of bits (digital code) from an arbitrary size input; the function is not reversible to produce the original input.

integrity — In information security, integrity means that data cannot be modified without authorization.

Internet Protocol Security (IPSec) — A protocol operating at network layer that defines encryption, authentication, and key management for TCP/IP transmissions.

intrusion detection system (IDS) — A tool to be installed at strategic locations (e.g. within DMZ) to monitor traffics continuously to look out for any suspicious activity.

Kerberos — A secret-key network authentication protocol, developed at MIT, that uses the Data Encryption Standard (DES) cryptographic algorithm for encryption, and a centralized key database for authentication.

message digest — The value returned by a hash function.

network address translation (NAT) — A function allowing a number of internal IP address (e.g. 10.X.Y.Z, 172.168.Y.Z) to share one or a few global IP addresses. This can not only allow more economical use of global IP addresses, but also hide internal computers from the Internet to lower their risks of being attacked.

non-repudiation — A property of a cryptographic system that prevents a sender from denying later that he or she sent a message or performed a certain action.

private key — A digital code used to decrypt information and provide digital signatures. The owner should keep this key secret; it has a corresponding public key.

proxy server — A software application that acts as an intermediary between external and internal networks, screening all incoming and ongoing traffic. It operates at the application layer.

public key — A digital code used to encrypt information and verify digital signatures. This key can be made widely available; it has a corresponding private key.

risk assessment — The process of identifying risks, quantifying the loss that may be resulted if such risks are realized, and determining the effective countermeasures to deal with the risks factoring in their costs and benefits.

RSA — The first full-fledged public key algorithm, named after its inventors Rivest, Shamir and Adleman, that can work for encryption and digital signing.

Secure Shell (SSH) — Providing secure remote connections to Linux/Unix hosts. Data is encrypted before being transmitted.

Secure Socket Layer (SSL) — A protocol that was developed by Netscape to provide a security sub-layer between application protocols (such as FTP, HTTP, or Telnet) and TCP/IP.

security audit — The process of examining the operation of networks and IT systems in great detail in order to determine whether they are as secure as they are supposed to be.

security policy — A means to let the management articulate how much importance they attach to computer security in their organization, and their directives to deal with it. It often comes as a statement to stipulate the security goals; risks; levels of authority; designated security coordinator and team members; responsibilities for team members; and responsibilities for each employee within the organization.

Virtual Private Network (VPN) — A means for establishing a secure network connection between two end points over an unsecure public or open network.

References

Dean, T (2012) *Network+ Guide to Networks*, 6th edn, Thomson Course Technology.

Kwan, R, Tsang, P, Kwok, P, Koong, K, Mak, J and Wu, J (2009) *A Practical Approach to Internet Programming and Multimedia Technologies*, Hong Kong: OUHK Press.

Network World (2009) '2008s biggest tech crime stories', http://www.networkworld.com/slideshows/2008/121508-year-in-cybercrime.html?netht=rn_121608&nladname=121608.