chapter 1 network security overview

1.1 security policy purposes

The

1.2 pillars of network security

The foundation of network security consists of:

- prevention
- detection
- response

Let's have a more detailed look at

1.3 security models

There are thre types of approaches in defining a security model:

- security by obscurity
- the perimeter defense model
- the defense in depth model

Security by obscurity relies on stealth for protection. The concept behind this model is that if no one knows that a network or system is there, then it won't be subject to attack. The basic hope is that hiding a network or at least not advertising its existence will serve as sufficient security. The problem with this approach is that it never works in the long term, and once detected, a network is completely vulnerable.

1.4 vulnerabilities, threats and attacks

Symmetric

1.5 vulnerabilities

A vulnerability

1.6 threats

Threat types:

- int

1.7 attacks

Attacks on the security of a system can be classified roughly into two categories:

- passive attacks
- active attacks

Passive attacks have the goal of intercepting and/or monitoring the traffic. There are two types of passive attacks – message interception and traffic analysis.

Active attacks imply modifying the normal flow of informations and/or creating illegitimate data

transmissions

If we classify the attacks by the the nature (scope) of these attacks, we can distinguish the following types:

- interceptions – an unauthorized entity gains illegitimate access to a network
- traffic analysis - the process of intercepting and examining messages in order to deduce information from patterns in communication
- interruptions – an element of the system is destroyed/incapacitated and becomes unusable or with considerable reduced capacity
- masquerading – requires an attacker to have the ability to both monitor and alter or inject messages into a communication channel
- alterations – an unauthorized entity may change the content of a data file or of a message exchanged over the network
- construction - an unauthorized entity may create and transmit false messages or may add new records in a data file or in a database

1.8 security services

A security system is supposed to provide the following services:

- authentication
- integrity
- confidentiality
- non-repudiation
- access control
- availability

Authentication allows the recipient of the message to validate the identity of the sender. It prevents an unauthorized entity to masquerade itself as a legitimate sender of the message.

Integrity guarantees that the message sent has not been modified or altered along the communication channel. This is usually accomplished by attaching to the message itself a digest (compressed version) of fixed length of the message, digest which allows verify if the original message was (intentionally or not) altered.

Confidentiality (secrecy) prevents unauthorized entities from accessing the real content of a message.

Non-repudiation with proof of origin assures the receiver of the identity of the sender, while non-repudiation with proof of delivery ensures the sender that the message was delivered.

chapter 2 security models

2.1 Par 1

A **cry**

Cipher

2.2 Par 2

2.2.1 Subpar 1

chapter 3 network security standards

3.1 Different security standards

While information security plays an important role in protecting the data and assets of an organisation, we often hear news about security incidents, such as defacement of websites, server hacking and data leakage. Organizations need to be fully aware of the need to devote more resources to the protection of information assets, and information security must become a top concern in both government and business.

To address the situation, a number of governments and organisations have set up benchmarks, standards and in some cases, legal regulations on information security to help ensure an adequate level of security is maintained, resources are used in the right way,and the best security practices are adopted. Some industries, such as banking, are regulated, and the guidelines or best practices put together as part of those regulations often become a de facto standard among members of these industries.

- **ISO STANDARDS**

The International Organization for Standardization (ISO), established in 1947, is a non-governmental international body that collaborates with the International Electrotechnical Commission (IEC) and the International Telecommunication Union (ITU) on information and communications technology (ICT) standards. The following are commonly referenced ISO security standards:

ISO/IEC 27002:2005 (Code of Practice for Information Security Management) ISO/IEC 27002:2005 (replaced ISO/IEC 17799:2005 in April 2007) is an international standard that originated from the BS7799-1, one that was originally laid down by the British Standards Institute (BSI). ISO/IEC 27002:2005 refers to a code of practice for information security management, and is intended as a common basis and practical guideline for developing organizational security standards and effective management practices.

ISO/IEC 27033-1:2009 - network security overview and concepts (see 3.2)

ISO/IEC 27033-2:2012 - Guidelines for the design and implementation of network security (see 3.3)

ISO/IEC 27033-3:2010 - Reference networking scenarios - threats, design techniques and control issues (see 3.4)

ISO/IEC 27033-4:2014 - Securing communications between networks using security gateways (see 3.5)

ISO/IEC 27033-5:2013 - Securing communications across networks using Virtual Private Networks (VPNs) (see 3.6)

ISO/IEC 27033-6: Securing wireless IP network access (see 3.7)

- **FISMA**

FISMA stands for Federal Information Security Management Act, and is a part of the USE-Government Act (Public Law 107-347) that became legislation in 2002 29. It requires US federal agencies to develop, document, and implement an agency-wide programme to provide information security for the information (and information systems) that support the operations and assets of the agency. Some of the requirements include:

1. Periodic risk assessments of information and information systems that support the operations and assets of the organization;
2. Risk-based policies and procedures designed to reduce information security risks to an acceptable level.
3. Plans for providing adequate security for networks and information systems;
4. Security awareness training to all personnel, including contractors;
5. Periodic evaluation and testing of the effectiveness of the security policies,procedures and controls.

The frequency should not be less than annually. Remedial action to address any deficiencies found to be properly managed;

6. A working and tested security incident handling procedure;
7. A business continuity plan in place to support the operation of the organization.

- **FIPS**

The Federal Information Processing Standards (FIPS) Publication Series of the National Institute of Standards and Technology (NIST) is an official series of publications relating to standards and guidelines adopted and made available under the provisions of the FISMA 30. FIPS Publication 199, Standards for Security Categorization of Federal Information and Information Systems, is the first mandatory security standard laid down under the FISMA legislation. FIPS Publication 200, entitled “Minimum Security Requirements for Federal Information and Information Systems” is the second mandatory set of security standards that specify minimum security requirements for US federal information and information systems across 17 security-related areas. US federal agencies must meet the minimum security requirements defined in this standard by selecting appropriate security controls and assurance requirements laid down in NIST Special Publication 800-53 (Recommended Security Controls for Federal Information Systems).

The 17security-related areas include:

1. access control;
2. awareness and training;
3. audit and accountability;
4. certification, accreditation, and security assessments;
5. configuration management;
6. contingency planning;
7. identification and authentication;
8. incident response;
9. maintenance;
10. media protection;
11. physical and environmental protection;
12. planning;
13. personnel security;
14. risk assessment;
15. systems and services acquisition;
16. system and communications protection;
17. system and information integrity.

Different FIPS standards:

FIPS 140 Security requirements for cryptography modules

FIPS 153 (3D graphics)

FIPS 197 (Rijndael / AES cipher)

FIPS 199 Standards for Security Categorization of Federal Information and Information Systems

FIPS 201 Personal Identity Verification for Federal Employees and Contractors

3.2 ISO/IEC 27033-1:2009 - network security overview and concepts

Revised and replaced ISO/IEC 18028 part 1;

Provides a roadmap and overview of the concepts and principles underpinning the remaining parts of ISO/IEC 27033;

Objective: “to define and describe the concepts associated with, and provide management guidance on, network security. This includes the provision of an overview of network security and related definitions, and

guidance on how to identify and analyze network security risks and then define network security requirements. It also introduces how to achieve good quality technical security architectures, and the risk, design and control aspects associated with typical network scenarios and network 'technology' areas (which are dealt with in detail in subsequent parts of ISO/IEC 27033). In effect it also provides an overview of the ISO/IEC 27033 series and a 'road map' to all other parts";

Provides a glossary of information security terms specific to networking;

Provides guidance on a structured process to identify and analyze network security risks and hence define network security control requirements, including those mandated by relevant information security policies;

Provides an overview of the controls supporting network technical security architectures and related technical controls, as well as non-technical controls plus other technical controls that are not solely related to network security (thus linking to ISO/IEC 27001, ISO/IEC 27002 and ISO/IEC 27005 plus other ISO27k standards as they are released);

Explains good practices in respect of network technical security architectures, and the risk, design and control aspects associated with typical network scenarios and network technology areas (expanded in subsequent parts of ISO/IEC 27033 - see below);

Briefly addresses the issues associated with implementing and operating network security controls, and the ongoing monitoring and reviewing of their implementation;

Extends the security management guidelines provided in ISO/IEC TR 13335 and ISO/IEC 27002 etc. by detailing the specific operations and mechanisms needed to implement network security controls in a wider range of network environments, providing a bridge between general information security management issues and the specifics of implementing largely technical network security controls (e.g. firewalls, IDS/IPS, message integrity controls etc.);

Mentions requirements such as non-repudiation and reliability in addition to the classical CIA triad (confidentiality, integrity and availability);

Somehow manages to provide a reasonably technical overview of network security with barely any reference to the OSI network stack!;

76 pages long;

Status: part 1 was published in 2009. It is currently being revised. The revision has easily passed the vote at DIS stage and may be published in 2015. Status update Jan 6

3.3 ISO/IEC 27033-2:2012 - Guidelines for the design and implementation of network security

Revised and replaced ISO/IEC 18028 part 2;

Scope: planning, designing, implementing and documenting network security;

Objective: "to define how organizations should achieve quality network technical security architectures, designs and implementations that will ensure network security appropriate to their business environments, using a consistent approach to the planning, design and implementation of network security, as relevant aided by the use of models/frameworks. (In this context, a model/framework is used to outline a representation or description showing the structure and high level workings of a type of technical security architecture/design)" [quoted from the FCD of 27033-1];

Defines a network security architecture for providing end-to-end network security. The architecture can be applied to various kinds of networks where end-to-end security is a concern and independently of the network's underlying technology;

Serves as a foundation for detailed recommendations on end-to-end network security;

Covers risks, design, techniques and control issues;

Refers forward to later parts of ISO/IEC 27033 for more specific guidance.

Status: part 2 was published in 2012.

3.4 ISO/IEC 27033-3:2010 - Reference networking scenarios - threats, design techniques and control issues

Objective is “to define the specific risks, design techniques and control issues associated with typical network scenarios” [quoted from the FCD of 27033-1];

Discusses threats, specifically, rather than all the elements of risk;

Refers to other parts of ISO/IEC 27033 for more specific guidance;

Status: part 3 was published in 2010.

3.5 ISO/IEC 27033-4:2014 - Securing communications between networks using security gateways

Revision of ISO/IEC 18028 part 3 and possibly ISO/IEC 18028 part 4;

Provides an overview of security gateways through a description of different architectures;

Guideline on securing communications between networks through gateways, firewalls, application firewalls, Intrusion Protection System [sic] etc. in accordance with a policy, including identifying and analysing network security threats, defining security control requirements, and designing, implementing, operating, monitoring and reviewing the controls;

Outlines how security gateways analyze and control network traffic through:

- Packet filtering;
- Stateful packet inspection;
- Application proxy (application firewalls);
- Network address translation NAT;
- Content analysis and filtering;

Guides the selection and configuration of security gateways, choosing the right type of architecture for a security gateway which best meets the security requirements of an organization;

Refers to various kinds of firewall as examples of security gateways. [Firewall is a commonplace term of art that is curiously absent from ISO/IEC 27000, ISO/IEC 27002 and is not defined explicitly in this standard either];

Status: part 4 was published in 2014.

3.6 ISO/IEC 27033-5:2013 - Securing communications across networks using Virtual Private Networks (VPNs)

Revision of ISO/IEC 18028 part 5;

Purpose: to provide “guidelines for the selection, implementation and monitoring of the technical controls necessary to provide network security using Virtual Private Network (VPN) connections to interconnect networks and connect remote users to networks”;

Extends the IT security management guidelines of ISO/IEC TR 13335 by detailing the specific operations and mechanisms needed to implement network security safeguards and controls in a wider range of network

environments, providing a bridge between general IT security management issues and network security technical implementations;

Provides guidance for securing remote access over public networks;

Gives a high-level, incomplete assessment of the threats to VPNs (i.e. it mentions the threats of intrusion and denial of service but not unauthorized monitoring/interception, traffic analysis, data corruption, insertion of bogus traffic, various attacks on VPN end points, malware, masquerading/identity theft, insider threats etc., although these are mentioned or at least hinted-at later under security requirements);

Introduces different types of remote access including protocols, authentication issues and support when setting up remote access securely;

Intended to help network administrators and technicians who plan to make use of this kind of connection or who already have it in use and need advice on how to set it up securely and operate it securely;

Status: part 5 was published in 2013.

3.7 ISO/IEC 27033-6: Securing wireless IP network access

Objective: “to define the specific risks, design techniques and control issues for securing wireless and radio networks” [quoted from the FCD of 27033-1];

This is a generic wireless network security standard offering basic advice;

WD3 lists a number of “threats” which are, in fact, attack modes or risks;

WD3 repeatedly refers to “access network”, a curious term that is not defined. It seems to mean “network” but without a definition, I cannot tell for sure;

WD3 indicates that encryption is a confidentiality and integrity control, whereas normally other cryptographic controls and protocols provide the integrity function, not encryption as such;

Status: at 2nd CD stage. The completed standard may surface by the end of 2015 but a request has been made to extend the project.

chapter 4 security policies implementations

4.1 Introduction

Information security has come to play an extremely vital role in today's fast moving, but invariably technically fragile business environment. Consequently, secured communications are needed in order for both companies and customers to benefit from the advancements that the Internet is empowering us with.

The importance of this fact needs to be clearly highlighted so that adequate measures will be implemented, not only enhancing the company's daily business procedures and transactions, but also to ensure that the much needed security measures are implemented with an acceptable level of security competency.

It is sad to see that the possibility of having your company's data exposed to a malicious attacker is constantly increasing nowadays due to the high number of "security illiterate" staff also having access to sensitive, and sometimes even secret business information. Just imagine the security implications of someone in charge of sensitive company data, browsing the Internet insecurely through the company's network, receiving suspicious e-mails containing various destructive attachments, and let's not forget the significant threats posed by the constant use of any Instant Messaging (IM) or chat applications.

4.2 Why have a security policy?

As building a good security policy provides the foundations for the successful implementation of security related projects in the future, this is without a doubt the first measure that must be taken to reduce the risk of unacceptable use of any of the company's information resources.

The first step towards enhancing a company's security is the introduction of a precise yet enforceable security policy, informing staff on the various aspects of their responsibilities, general use of company resources and explaining how sensitive information must be handled. The policy will also describe in detail the meaning of acceptable use, as well as listing prohibited activities.

The development (and the proper implementation) of a security policy is highly beneficial as it will not only turn all of your staff into participants in the company's effort to secure its communications but also help reduce the risk of a potential security breach through "human-factor" mistakes. These are usually issues such as revealing information to unknown (or unauthorised sources), the insecure or improper use of the Internet and many other dangerous activities.

Additionally the building process of a security policy will also help define a company's critical assets, the ways they must be protected and will also serve as a centralised document, as far as protecting Information Security Assets is concerned.

4.3 What is a security policy?

The security policy is basically a plan, outlining what the company's critical assets are, and how they must (and can) be protected. Its main purpose is to provide staff with a brief overview of the "acceptable use" of any of the Information Assets, as well as to explain what is deemed as allowable and what is not, thus engaging them in securing the company's critical systems.

The document acts as a "must read" source of information for everyone using in any way systems and resources defined as potential targets. A good and well developed security policy should address some of these following elements:

- How sensitive information must be handled
- How to properly maintain your ID(s) and password(s), as well as any other accounting data
- How to respond to a potential security incident, intrusion attempt, etc.
- How to use workstations and Internet connectivity in a secure manner

- How to properly use the corporate e-mail system

Basically, the main reasons behind the creation of a security policy is to set a company's information security foundations, to explain to staff how they are responsible for the protection of the information resources, and highlight the importance of having secured communications while doing business online.

4.4 Getting started

The purpose of this section is to provide you with possible strategies and some recommendations for the process of creating a security policy, and to give you a basic plan of approach while building the policy framework.

The start procedure for building a security policy requires a complete exploration of the company network, as well as every other critical asset, so that the appropriate measures can be effectively implemented. Everything starts with identifying the company's critical informational resources, a subject that is discussed in depth in the next section of the paper.

4.5 Risk analysis (Identifying the assets)

As in any other sensitive procedure, Risk Analysis and Risk Management play an essential role in the proper functionality of the process. Risk Analysis is the process of identifying the critical information assets of the company and their use and functionality -- an important (key) process that needs to be taken very seriously. Essentially, it is the very process of defining exactly WHAT you are trying to protect, from WHOM you are trying to protect it and most importantly, HOW you are going to protect it.

In order to be able to conduct a successful Risk Analysis, you need to get well acquainted with the ways a company operates; if applicable, the ways of working and certain business procedures, which information resources are more important than others (prioritizing), and identifying the devices / procedures that could lead to a possible security problem.

List everything that is essential for the proper functionality of the business processes; like key applications and systems, application servers, web servers, database servers, various business plans, projects in development, etc.

A basic approach would be:

- Identify what you're trying to protect
- Look at whom you're trying to protect it from
- Define what the potential risks are to any of your Information Assets
- Consider monitoring the process continually in order to be up to date with the latest security weaknesses
- A possible list of categories to look at would be:
    - Hardware: All servers, workstations, personal computers, laptops, removable media (CD's, floppies, tapes, etc.), communication lines, etc.
    - Software: Identify the risks of a potential security problem due to outdated software, infrequent patches and updates to new versions, etc. Also take into account the potential issues with staff installing various file sharing apps (Kazaa, Sharereactor, E-Donkey, etc.), IM (chat) software, entertainment or freeware software coming from unknown and untrustworthy sources.
    - Personnel: Those who have access to confidential information, sensitive data, those who "own", administer or in any way modify existing databases.

4.6 Risk Management (Identifying the threats)

Based on the research conducted on the company's information assets, you should now be able to properly manage all the threats posed by each of your resources.

The purpose of this section is to guide you through the creation of a list outlining various potential threats, something that should also be included in the formal security policy. Each of the following elements will be discussed in depth later in the Security Awareness Program section, thus providing the staff members with a better understanding of each of the topics covered below.

4.6.1 Physical/Desktop Security

- **System Access**: best practices for password creation, passwords aging, minimum password length, characters to be included while choosing passwords, password maintenance, tips for safeguarding (any) accounting data; the dangers to each of these issues must be explained in the security awareness program;
- **Virus Protection**: best practices for malicious code protection, how often the system should be scanned, how often, if not automatically, should Live Update of the software database be done, tips for protection against (any) malicious code(viruses/trojans/worms);
- **Software Installation**: is freeware software forbidden, if allowed, under what conditions, how is software piracy tolerated, are entertainment/games allowed or completely prohibited as well the installation of any other program coming from unknown and untrustworthy sources;
- **Removable Media(CD's, floppy)**: "Acceptable Use" measures (perhaps by way of a AUP – Acceptable Use Policy) need to be established, the dangers of potential malicious code entering the company network or any other critical system need to be explained as well;
- **Encryption**: explain when, how and who must encrypt any of the company's data;
- **System Backups**: the advantage of having backups needs to be explained; who is responsible, and how often should the data be backed up;
- **Maintenance**: the risks of a potential physical security breach need to be briefly explained;
- **Incident Handling**: define what a suspicious event is, to whom it needs to be reported, and what further steps need to be taken;

4.6.2 Internet Threats

- **Web Browsing**: define what constitutes restricted, forbidden and potentially malicious web sites, provide staff members with brief, and well summarized tips for safer browsing, additionally let them know that their Internet usage is strictly monitored in order to protect company's internal systems;
- **E-mail Use**: define the "acceptable use" criteria of the E-mail system, what is allowed and what is not, the company policy on using the mail system for personal messages, etc. Also briefly explain the potential threats posed by (abusing) the mail system and of the potential problems as far as spreading malicious code is concerned;
- **Instant Messaging (IM) Software (ICQ, AIM, MSN, etc.)**: whether it is allowed or completely forbidden, provide them with short examples of how an attacker might use these programs to penetrate and steal/corrupt/modify company data;
- **Downloading/Attachments**: is downloading allowed or not, useful tips for safer downloading, explanation of trusted and untrustworthy sources, best practices for mail attachments if allowed, discussion of potential threats and dangers, use of virus scanners, etc.

These elements will later be covered in detail in a Security Awareness Program. Staff need to understand why some activities are prohibited, what the impact of certain dangers can have on the company, actions they must follow if and when a potential security problem has been suspected or discovered. By involving staff in a Security Awareness Program staff will not just broaden their knowledge on the information security field, but also learn how to act in a secure manner while using any of the company's information assets.

4.7 Security policy violation

In order to realise the importance of a security policy, staff need to be aware and fully understand the

consequences of violating the policy, thereby exposing critical systems to a malicious attacker, or causing unintended damage to other companies worldwide. Violations should be handled accordingly; those who in one way or the other violate the security policy should be made aware that they may face being put through a "trial period", which involves also the limited use of some of the company information assets until they can show they are able to act in a secure manner while using the corporate systems. They should also be aware that in some (severe) cases they also may risk being fired or even prosecuted.

Whereas this may seem as overkill to some, appropriate action needs to be taken in every violation case in accordance with the terms of the AUP and the policy, with the focus on reiterating the security basics and not punishment. Otherwise there will most likely be a successful penetration, either due to human error, or misunderstanding the policy.

4.8 The implementation of the policy

When the security policy is all drawn up, revised, updated and agreed upon, the implementation process will follow. This is usually harder than the creation of the policy itself, due the fact that at this stage you also need to coach and educate your staff to behave in a "secure" manner, following each of the core elements pointed in the formal security policy.

The final version of the security policy must be made available to all of your employees having access to any of your information assets. The policy must be easily obtainable at any time, with a copy placed on the internal network and intranet, if applicable.

A proper implementation requires not only educating staff on each of the core elements flagged as critical in the formal Security Policy, but also changing their role in the effort to protect critical company data.

The next section will aim to guide you through the creation process of a basic Security Awareness Program, along with various innovative and interesting ways of educating your staff, using user-friendly & informal lines of communication between the Information Security Office (ISO) members and your employees.

4.9 The process of developing

This section will provide you with the various strategies of building a solid Security Awareness Program. We will discuss various methods, their advantages and disadvantages, and will also give you get a better understanding of the essential steps to building the Program.

At the beginning you must answer yourself the following questions:

- What is the Security Awareness Program supposed to accomplish, and how are you going to draw attention on that?
- Who is your audience, how "educated" they are; is it going to be necessary to divide the program into two parts, one for those who have more knowledge about computers, and one for those who are not much into computers at all?
- How are you going to reach and motivate your audience? More importantly, how are you going to get your audience interested in improving the Information Assets of the company?
- Is the Program going to rely on a formal or an informal way of communication between you and the staff members? In which way are you going to conduct and present it?

4.9.1 The Purpose Of The Program

First of all, you need to explain to staff what the program will be trying to accomplish, how it will aim to improve the operations of the company, and how vital the protection of Information Assets really is. You will need to explain why "Security is everyone's responsibility", and ensure everybody understands it; explain that even if the company has the latest technological improvements like firewalls, intrusion detection systems, etc., an uneducated staff member could easily endanger sensitive information, and render any technical security measure in place, completely and utterly useless.

Another common misunderstanding that you will definitely face while conducting the Program is that the

majority of people often tend to think that it is not their responsibility to help improve the security of their company. Generally people are of the (wrong) opinion that only the IT department or Information Security Office(ISO) can and need to take care of issues like these, and that is where generally the buck stops.

4.9.2 Addressing The Audience

One major problem that I am sure you are going to be facing is the difference in the levels of computer skills (of your audience), which will sometimes force you to pay additional attention to those who are not that much into computers. On the other hand you could also choose to differentiate between those who need security education, and those who don't; the idea is to separate staff having access to any of the company information assets from those who don't (and can't endanger sensitive data in any way), as this will definitely save you a lot of time and resources. It would be a good approach to hold informal meetings with staff in order to talk on a personal level and also conduct several surveys in order to measure their skill level; this way you will know where to focus your attention to.

4.9.3 Measuring Their Security Awareness Level Through Surveys

Security Awareness surveys are developed with the idea of measuring the current Awareness level of your staff, but will usually also point out common mistakes and misunderstandings of your employees; which will definitely help you improve the quality of the Program, even before it starts. It is highly recommended to archive the surveys in order to evaluate the effectiveness of the Program over a period of time.

You might also want to indicate to staff members that the survey is completely anonymous, that there is no need to cheat as the main idea is to merely measure the overall security awareness level in the company, and above all that this is just a survey and not an exam. They could answer just the main question without having to answer the "Why do they think so" section, if they don't know what to give here as an answer.

Some sample Security Measuring Survey questions might be:

1. Which of the following passwords is the most secure one, and why do you think so?
    - Abc123456
    - HerculeS
    - HRE42pazoL
    - $safe456TY

    Why do you think so?
2. Which is the most dangerous attachment extension, and why do you think so?
    - *.exe
    - *.com
    - *.bat
    - *.vbs
    - all of the above

    Why do you think so?
3. Your security policy states that the Information Security Office (ISO) would never send you an update to an application, but you have just received one, what would you do next?
    - as it's coming from security@company.com which is our ISO e-mail address I will just run it and have the latest version of the software.
    - as stated in the Security Policy I need to scan all the attachments before running, so I will scan and run after that.
    - I would call the ISO office immediately to request further information.

4. A friend of yours gave you a multimedia CD last night, which you intend to check from your workstation at work; how are you going to do it?
    - he is a friend of mine, and he would never give me any destructive files like viruses, etc. I trust him/her that's why I am going to check it out right away.
    - although he is a friend of mine, it is stated in the security policy that removable media is allowed but its use should be limited to the minimum; I will stick to that and would scan the CD contents and see what's inside before I do so.
    - I would just check the contents of the CD from my personal PC.
5. An ISO office representative asks you (in person) for your password as they misplaced it, and would need it to implement further security measures on your workstation; what would you do?
    - they can't access the workstation without my password, and as it is about improving security, I would give it to them, as they are those responsible for maintaining the security within the organization.
    - I already have the workstation properly secured so I won't give it to them.
    - I won't share my password with somebody even if my manager tries to force me into telling it; I would keep it as secret as possible.

These are some sample questions covering most of the threats pointed out in the Security Policy. It is completely up to you to decide how many questions should be in the survey as well as the aspects they should cover; but it is advisable to consider issuing surveys on a regular basis in order to continuously monitor the level and the effectiveness of the Program.

4.9.4 Getting Their Attention

Staff already have a lot of things to think about, a lot of decisions to make, operate and run through most of the day-to-day business procedures; therefore you need to have a very good strategy to get them motivated & eager to learn how they can improve company security.

Everyone these days is interested in stories about computer security in one way or the other, especially the (high-profile) break-ins, and making use of this, your main aim will be to help understand "attendees" of the program that they are actually going to be the new "gatekeepers" of critical company data (the information assets). You will undoubtedly will get asked questions like "Yeah, it's great to contribute to company security, but what do I get in exchange", which I define as normal questions, to which you must give proper answers.

Your future "students" need to be made aware of and understand how expensive it is for a company to conduct Security Awareness Courses, and to employ security experts, in order to provide 'attainable' services to its customers. Explain to them the damages that could be inflicted to the company, to the company (brand) name, its image, etc., which will inevitably impact on them somehow in return.

On the other hand, draw their attention also to the personal benefits from the whole program and the value of all the knowledge that will be supplied to them. One good example is to mention how all that information will significantly help them increase the security level of their own personal computers at home. The information they will be provided with does not only apply for their PC's at work but applies (in full) for their home PC's as well.

Another important point to keep in mind is the different ways people learn and memorise things, or in other words, deal with information they have just been provided with. Some learn by reading the materials, while others learn more by looking at diagrams, although it is proven that a combination of these methods has maximum effect in the process of understanding the subject. Therefore, you must ensure that your presentation style is such that it appeals to a crowd of people with varying degrees of knowledge and understanding.

Everyone gets bored of reading long materials, no matter how interesting they might be; if there isn't a picture, diagram or anything that brings some sort of variety into the process, people leave it behind. Try to "visualise" every subject that you are talking about by adding plenty of pictures, diagrams, relevant artwork and cartoons.

Cartoons are especially good, as they add an element of humour; people will definitely remember a funny situation representing a far serious procedure. Cartoons are best suited for posters, and most effective when placed all over the company, with their main purpose being a friendly medium to spreading the security awareness program messages (i.e. "Lock your machine when you leave", or "don't share your ID and password with ANYONE", etc).

Humor plays an essential role in the friendly education of the staff members; consider using it as you see fit but don't turn the whole program into a big comedy where everyone laughs and just makes jokes about the word Security. The addition of a little, humorous anecdote to any of your lectures like "I've got a friend who's so paranoid about security that he burns every paper after work, but come on, don't set the fire alarms off, just shred those papers labeled confidential/secret" would do fine.

4.9.5 Choosing The Approach

There are several approaches that you can follow when educating staff, and this section will point out the one which I define as the best one; a combination of both formal and informal ways of education.

The advantage of the formal method is that it will help staff realize the very importance of the security issue, as they know that these presentations cost a fair amount of resources, effort and money. On the other hand it will highlight the fact that the company is taking security very serious and therefore taking very serious measures to protect its information assets by educating its staff; and all it requires from them is a little time, devotion and understanding the importance of the security issue.

Another highly beneficial point when conducting a formal Awareness Program is the fact that your message, tutorial, presentation will be spread between most, if not all of the staff members; you will reach a lot of people that way, which will save you a lot of time compared to methods like one-on-one sessions, etc.

The informal way of education consists of email reminders, discussions, posters spreading security oriented messages (that are mostly discussed at the Course), screen savers, mouse pads, mugs, stickers, etc. as Security Awareness directors keep finding new and innovative ways of educating staff members. The advantage of this method is that it doesn't push (or, oblige) people in any way, like attending a meeting, listening to lectures, etc.; it is very personalized, user friendly and highly effective due to the fact that it comes very close to their every day life and working procedures within the company (posters, mouse pads, etc.).

Informal discussions are another highly beneficial way to educate and measure the skills of staff where people ask questions, answered by a representative from the ISO; the atmosphere is usually much more informal and calm. This is a highly recommended way of communicating with employees, as it initiates a two-way conversation whereby many points can be covered.

As in many other aspects, you need to find the right balance between the formal and informal ways, as both of these methods have their various advantages and disadvantages. By closely monitoring reactions from staff to the meetings and lectures conducted, you will be able to significantly revise and continuously improve the quality of your Security Awareness Program. Always provide staff with an always-evolving way of education, thus keeping them interested, eager to know and learn, and reducing the chance of boredom, while attending any of the Program's events.

4.10 Security threats management

Once you have defined the best way for education, have your plan and strategy ready, measured the computer skills level of your staff, you should start by discussing each of the elements pointed out in the security policy, in detail.

The main purpose of this section is to explore each of these elements in detail and to discuss various threats, providing you with ready made "Best Practices" on various topics along the way. You are encouraged to include parts of this section in your own Security Awareness Course(s), thus providing your staff with a better understanding of the issues covered below.

4.10.1 Physical/Desktop threats explained

The threats that will be discussed in this section concern the way you use your workstation, access

restricted zones in the company, and the way you handle sensitive information. I will cover all the possible threats, discuss their importance in detail, and provide you various effective ways to manage them.

4.10.2 System access

Staff need to be fully aware of their responsibility to keep their User ID and password as secret as possible, and it's all because this is the first line of defense within any system: the identification of the user. Explain to the user that it is completely forbidden to share his/her ID and password with ANYONE, by ANYONE you mean, anyone ranging from the representatives of the Information Security Office (ISO), to their family members. No matter how stupid this might sound to some, they must not do it; even if their manager asks them for their password, they must reject the request. This way, NO ONE can force them into revealing their ID and password, under any circumstances. I know of cases where managers have tried to force (or even, trick) their staff into giving out their passwords for some reason or other in order to evaluate their level of security awareness; to see if they comply with what is stated in the Security Policy, i.e. not to share their ID and password with ANYONE. It is always useful to provide personnel with such "live" examples of how their awareness might, and is being evaluated.

Staff are required not to write any accounting data or ID/password information on loose papers, or sticky (post it) notes, or leave sensitive information on white boards (for example, after a meeting, white boards, and/or flip charts should be cleared off) as this could result in a potential break-in, due to the improper handling of sensitive data. No matter how safe staff might think their password is, they should not be allowed to store them on any of these bits of paper; they must do their best and memorize it instead. Another common mistake that must not be overlooked is the horrifying fact that most of the users tend to hide these notes under the keyboard, or on some "secret" place, as they call it, around their desk; another activity that should be completely forbidden due to obvious reasons. Someone could easily find the "secret" hiding place and get acquainted with vital accounting data.

You must also educate your staff in the way that strong passwords are created. The (secure) ways accounting data must be handled are outlined in the "Password Best Practices" document, which briefly summarizes these two aspects. I have included a sample "Password Creation Best Practices", and a sample "Password Maintenance Best Practices" section below, which will give a overview of what must be taken into account while writing such documents.

4.10.3 Password creation best practices

- Passwords must be made up of a mixture of lower-case (small) letters, upper case (capital) letters, numbers, and at least one special character, such as (!@#$%^&*()_+|);
- The minimum length of the password must be at least 7 characters;
- Do not use the same password on several computers and/or services as once revealed, it would compromise the security within all the others in one go.

Good Examples

- Ona327(sA
- @865DapzI
- 93Sow#-aq

All of these are examples of good passwords, because they fully comply with Password Creation Best Practices; thus containing a mixture of small letters, capital letters, as well as numbers and special characters.

Bad Examples

- aaa123bbb
- abcdefg
- 76543210

The first is a terrible one, and any properly configured cracking program will retrieve it in a matter of minutes, and let's not even mention the second and the third one. The user with the last password (76543210) obviously thought it would be an easy to remember password, as well being a secure one, as it is a long(ish) one; but what the user does not know or realise is the fact that most cracking programs will find it in a matter of seconds (as the password follows a specific numerical pattern). It might be a good idea to incorporate a little demonstration in your Awareness Course at some point providing your staff with the unique opportunity to see how a (password) cracking software operates.

Strong Passwords Creation Tips

- Use the first letters of a quote, song, etc., for example "Something takes a part of me..." would be 'Stpm'
- join two words, include a number, as well as a special character, for example 'run4life#';
- a nice strategy when memorizing passwords might be the following:

  Let's assume your password is Naige453$lZ; first, pronounce it several times in your mind, then ask yourself what your password is, answering this question in the following way: "My password is a mixture of the name Naigel (a foreign friend of mine), several numbers and a dollar sign; my password starts and ends with capital letters, before the last letter of the name (L) there is a dollar sign ($), and before the dollar sign, there are random numbers.

  This is a very useful and helpful trick for anyone who is trying to memorise or remember their password. By repeating (almost explaining) to yourself what your password is describing it the way I suggested above, I am certain that you will not have any problems remembering sophisticated, yet strong passwords.

4.10.4 Password maintenance best practices

The proper maintenance of sensitive data such as the User ID and password are a responsibility of every staff member. This section will briefly cover Password Maintenance Best Practices.

- Do NOT share your User ID(s) and password(s) with ANYONE, neither with an ISO representative, help desk staff, family members, nor with your manager(s). No one can force you into revealing your User ID(s) and password(s) under any circumstances, remember that. It is your responsibility to keep the data as secret as possible;
- Do NOT store your User ID(s) and password(s) on any loose bits of paper, sticky (post-it) notes, white boards, flip charts, etc.;
- Do NOT hide your User ID(s) and password(s) under the keyboard, or at any other would be "secret" hiding place. Do your best and memorise it;
- Do change your password(s) following the stated password renewal period in the security policy;
- Before entering your User ID and password, make sure no one is watching you, to avoid the so-called "shoulder surfing" technique.
- Before using your User ID and password on a third-party computer, make sure it is well protected, and free of trojans and key loggers.

4.10.5 Virus protection

Based on published papers, expert's predictions, as well as drawing upon personal experiences, I can easily state that viruses will continue to be a very serious threat to critical business data, and will continue to evolve, becoming more sophisticated, dangerous and devastating.

When you start explaining what a virus is, limit it to the facts, for example like how destructive it is, what damage it can cause, the possible financial losses related to a virus outbreak, etc. Don't bother staff with the specific technical information such as ways viruses function, how they hide, and many other topics that will not be of interest to them. Instead, provide those who are most interested, with some external (internet) links to the subject.

Consider explaining what a virus/trojan/worm is, the basic functions of each of these, how to possibly recognize (the operation of) one on your system(s), the potential problems they could cause, and the devastating effects on the whole company. Provide them with live examples, briefly discuss and answer the most simple and frequent questions that come up such as "Can the data corrupted from viruses be recovered", or "What to do once infected with a virus". However, you need to clearly explain that the idea of the presentation is to prevent an infection in the first place, as once infected with a destructive virus, there is not so much you can do, especially if there are no backups of the data.

On the other hand you need to precisely explain what the personal damages would be after a virus infection; damage and/or potential loss of critical business data, documents, projects, business plans, presentations they have been working on, along with any other personal data stored on the computer will be damaged, or, more than likely, be destroyed. By getting to know the devastating effects that viruses may have, staff will be much more aware on the subject, and will more than likely understand the importance of the topic, and the risks for both their company and their home PC's.

Go through the many scenarios of how viruses can get into the company networks, how staff could be tricked into running a virus, the dangers posed by Internet downloads, problems with outdated virus signatures, etc. Also explain the fact that Anti-Virus (AV) scanners are not the best, "fool-proof" solution, and the way they rely on signatures (pattern files). Discuss how useful virus scanners are, and how the effectiveness of preventive measures that are in place depend for a large part on the awareness and vigilance of the users themselves.

Staff need to understand that our main aim is to try and prevent, not to act after we are infected; although there will definitely be infections, we can significantly reduce the risk of infection and limit potential damage by educating staff and making them aware of the dangers posed by malicious code and software (virus/trojan/worm).

The advantages of regular system scanning, as well as the potential problems of not scanning your systems need to highlighted as well; although they know that AV scanners will not detect new viruses, they will at least know that they can reduce the risk, and properly manage the danger.

Scanning the systems using outdated signature files is another common problem that need to be touched upon. Staff should update their Anti-Virus/Anti-Trojan software at least once a week, and if the software allows centralized automatic updates (most do), updates must be scheduled on a regular basis to ensure the software detects the latest viruses/trojans/worms (known to your vendor's lab).

The various ways of getting infected with malicious code should be highlighted as well; let the employees openly ask you questions: see how they react to questions like "How am I getting infected", and then provide them with a better or more complete explanation about the most common as well as specific ways of infections. Below, I have included a sample "Malicious Code Best Practices" section for your convenience ; by no means an exhaustive list, but at least you will be able get an idea of what is considered as dangerous activity.

4.10.6 Malicious code best practices

- Do NOT run any files without first scanning them, no matter what the file extension is, i.e. (.exe, .bat, .com, .doc, etc.);
- Do NOT download any files and/or programs from unknown sources; if in doubt, contact the ISO office as soon as possible;
- Do NOT open attachments, even if they were sent by a friend or family member; verify first that indeed, he/she has sent you the file, but nevertheless scan before you open/run anything;
- Do NOT run any programs you have found on diskettes/CD's around your desk if you are not completely sure that they are yours; someone might have placed it there specially for you to "find it and check it out";
- If downloading is allowed, limit it to the minimum; if you need a specific application or something else, always contact the IT department or the ISO office for further information BEFORE you download and installing something;
- Scan (full system scan) your system at least once per week with your default AV scanner software.

Be sure to update the virus signatures before you do so, and also consider automating the process by scheduling a Full System Scan for convenient regular scanning in the future;

- Update the signature files as often as possible, so to ensure that the latest malicious software patterns are detected;
- The IT department or the Information Security Office will usually NEVER mail you the latest updates of any software (unless this is preceded by a much publicised, well-advertised, company wide campaign). If you detect suspicious activity, do not delete the e-mail received and contact the Incident Handling or Help Desk team as soon as possible;
- if you have any doubts regarding malicious software (viruses/trojans/worms), contact the ISO, The Help Desk or the IT department immediately. This way you will prevent any potential devastating mishaps, due to inappropriate and erroneous handling of dangerous and harmful incidents.

Everything that is defined as forbidden must be discussed and explained; why it is forbidden or restricted, how it could harm the company or the business, etc. Play out several potential scenarios, thus helping the users grasp the topic in an easy to understand way while trying to touch base on the consequences of all of these dangerous activities.

4.10.7 Software installation

Freeware, or any other type of software, obtained or downloaded from unknown or untrustworthy sources could easily affect company security, exposing critical business data and/or corrupting sensitive ones. A lot of users tend to install such programs (from screen savers to games and funny cartoons in Flash) as they put it, for various personal needs and activities; to entertain, have something nice to look at or relax themselves. At the same time, they do not realise the potential threats they are exposing the company systems and networks to, from malicious software (viruses/trojans/worms) to legal actions against the company for installing (possibly) pirated software on the company workstation(s). Thus, you need to familiarise users with the potential problems attached to each of these issues, and also explain the company policy towards installation of any (unauthorised) software on any of the company workstation(s). Files downloaded from the Internet, copied from a CD or a floppy coming from an unknown source, or anything else that has not been reviewed by the Information Security Office or not been scanned for potential malicious code (by the corporate AV systems) could actually be classified as untrustworthy, unknown and dangerous. Freeware applications, due to their nature of origin, are a significant source of threat and should be approached with caution.

Staff members need to be aware of the risks involved, and learn to think twice before they act on issues. This can be stimulated in many ways; by playing out various scenarios on how software downloaded from either the Internet or copied from any removal media could endanger the company, its business, one's privacy, or the use of company bandwidth to commit illegal actions.

It is entirely up to you to decide whether users should be allowed to download and/or install third party programs on their workstations; and implement the appropriate (security) policies and procedures that go with this decision. You will not only need to clearly state the consequences for those who violate any restrictions, but also provide the procedures for obtaining and installing new software.

It is highly recommended that users do not have the ability to install any new programs that might either expose sensitive company information, waste valuable bandwidth, or corrupt critical data. If users need new software installed for business use, they should contact their manager, the IT/IS department, the Help Desk or the ISO (depending on the procedures set out in the policy) instead of undertaking such action themselves.

4.10.8 Removable media (CD's, floppies, tapes, etc.)

Removable media such as CD's (Compact Disks), floppies (Floppy Disks) and even tapes (backup/ADR/DAT/DLT tapes) can be defined as another possible entry point for dangerous and malicious files entering the company network or endangering the security of a single workstation. On the other hand, these can also be used to illegally copy sensitive data on, after which it would be easy to walk out of the premises with the stolen information.

Malicious software (viruses/trojans/worms) also use removable media to spread; some take advantage of the auto-run feature of the CD (automatically executing the auto-start file on the CD, which could be a

destructive one), others still use "classic" methods like diskettes to get the workstation infected with a malicious program. For best results, removable media devices should be banned entirely (utilising floppy drive-locks, or 'CD-less' workstations; CD's can still be used via CD-Towers, for example). If you need to use removable media in your organisation, then best practices must be established, possible risks and danger scenarios discussed so to reduce malicious programs entering your networks at all points, thus protecting your company from a major disaster.

4.10.9 Encryption

Encryption can be defined as another "must implement" measure that will not only keep your sensitive and critical information secured against a potential attacker, but also protect you from a lot of trouble if eventually a security breach does occur. In your security policy and procedures you must clearly define the systems, files and documents that should be encrypted, by whom, and most importantly, using which algorithms. It is strongly recommended to use proven, industry standard algorithms, such as DES, IDEA, Blowfish, or RC5.

4.10.10 System backups

Disaster recovery (DR) plans are essential for the continuity of the business as well as the proper functionality of the current processes. Sooner or later you will inevitably face the problem where a system crashes, no matter of the OS used, but this can be dealt with promptly, if proper backup procedures and disaster recovery plans are in place.

You will have to define the assets that must be backed up on a regular basis, the responsible individuals, best practices and procedures, as well as where the backups should be stored, i.e. a fireproof safe, vault, off-site, etc.

4.10.11 Maintenance

The proper maintenance of the PC/workstation is another vital issue that must not be overlooked during the course of the Security Awareness Program. Users' workstations are a significant source of threat to company security, often targeted by the so-called "insiders" snooping around, looking for unprotected workstations. Therefore, you need to educate staff on the aspect of Physical Security as well; again, this can be achieved by running through the possible scenarios, while providing tips for better overall protection.

4.10.12 Incident handling

By now your staff should be able to define a potential security problem, while you should be establishing the rules for the course of action to take in case of an incident. In your policy you must clearly state what must be done in various situations; the main idea here should be to minimise and limit damage. Staff should be made aware who is responsible for handling problems, and whom they should contact as soon as they suspect a potential security problem.

4.10.13 Internet threats explained

One of the greatest security risks in the company is the Internet connectivity, and its misuse through (uneducated) employees. It is a fact that most employees will surf to sites that are strictly prohibited, and most probably will end up downloading malicious files and/or hostile code from hacker sites somehow. Any of these activities could impact the productivity of your company, especially if you think about the recovery process trying to rectify the mistakes made by staff.

Therefore, it is always a good idea to explain in detail the possible dangers of surfing the Internet; that you don't need to download anything at all to get the computer infected with a virus, trojan or even a worm but just visiting the site is enough to cause a problem. Define what constitutes a "prohibited site", and explain why it is prohibited, including the problems that could occur just by visiting it.

4.10.14 Web browsing

Web browsing represents a threat to the security of the workstation, as well as to the whole organisation. Being exposed to the dangers of web browsing is very easy as hostile scripts could be downloaded, and executed automatically; all it takes for example is an outdated version of the web browser.

Staff should be able to make a distinction between sites that are classified as allowed, prohibited or potentially dangerous, and try avoid visiting prohibited ones. Java and ActiveX should be disabled by default (it will not give problems accessing pages), care must be taken with Flash movies, etc. and if ever a hint of a problem occurs, the ISO office must be contacted immediately.

There are web sites in the wild, that could attempt to scan/flood your network, just by visiting them; another variant to this (theoretical, but very possible) scenario is one of your employees using some kind of scanning service to check the security of his/her workstation, thus wasting valuable bandwidth. Something like this will invariably produce more work for the ISO office as well as their systems probably will register the usage of this service as a possible break-in attempt. Online gambling and pornographic web sites should be fully prohibited, and the web usage of staff monitored to ensure they are following the rules and regulations set forth by the Security Awareness Program.

4.10.15 E-mail use

Generally the company E-mail systems are a high risk area due to their constant availability to the outside world, and the risk is often two-fold. The use of e-mail to conduct business, contact clients, and its integration in many other business related processes exposes company mail addresses and (mail) systems to potential attackers. On the other hand, this is also the number one entry point from which most of the malicious programs are entering the company. Therefore, a well-known and proven malicious code protection program is a must have on all the mail gateways, as it will detect, block and/or filter out most of the known dangerous files and hostile scripts trying to enter the company networks.

As with all aspects of IT security, company-wide security can only be improved through the proper education of staff. It is therefore highly recommended that you establish Best Practices for E-mail use, concentrating on the points below.

4.10.16 E-mail use best practices

- If (E-mail) attachments are allowed, the attachment(s) must be scanned before opening as well as confirming with the sender (i.e. via phone) that indeed an attachment has been sent. This will also reduce the risk of running a program that has been e-mailed out automatically (unknown to the originator) via some kind of malicious application that has made use of the mail account(s) and/or mailing system of the sender. If attachments are forbidden, follow the policy and do not download/run any file(s) received as attachments;
- Java and ActiveX must be disabled while reading e-mail in order to manage the risk of auto-executing malicious programs. Just like in the internet browser, certain options of the program can usually be set and locked by way of system policies that automatically set these conditions for all users at logon;
- Do not use the company e-mail accounts for registration purposes of any kind, and do not use it while posting messages in web forums or newsgroups. You may want to create one, special (possibly aliased) account for this purpose only;
- Do not use the company e-mail system for running your own business, excessive personal mailing, sending large attachments, thus wasting valuable bandwidth;
- Do not respond to chain letters, or any other sort of spam using the company e-mail systems; if in doubt, contact the ISO office;
- Never forward any company data to external e-mail accounts (i.e. send a work document to your home email account, so to work on it further from home that evening), without first checking with your manager and/or contacting the ISO office;

- The proper use of the E-mail system should continuously be monitored and the users should be aware that they could be held liable for illegal activities, such as spamming, sending and receiving illegal content, etc.

4.10.17 Instant messaging (IM) applications (ICQ, AOL, MSN, etc.)

A lot of users tend to use these programs in order to communicate with friends, send and receive attachments, messages, etc as these applications often try to trick the content blocking gateway at the server level to letting content pass through. However, they do not fully realise the dangers of these programs, and the potential damages that they could cause.

A snapshot from our previous publication 'The Complete Windows Trojans Paper', available from our web site at http://www.frame4.com/publications/index.php, reviews various scenarios of getting infected with a malicious program via ICQ:

- You can never be 100 percent sure who is on the other side of the computer at that particular moment. It could be someone that hacked your friend's ICQ UIN (Unique Identification Number) and wants to spread some trojans;
- Old versions of ICQ had bugs in the WebServer feature that creates a web site on your computer with your info from the ICQ database. The bug meant that the attacker could have access to EVERY file on your machine ... and you probably realise what could happen if someone has access to your win.ini or some other system file: a trojan installed on your computer in a few minutes;
- Trojan.exe is renamed to Trojan....(150 spaces).txt.exe, the icon changed to a real .txt file; this will definitely get you infected. This bug will most probably be fixed in the newer versions.

No matter the Instant Messaging application you are using, you could always get infected or exploited; through a specific application bug you never heard about or a buggy version you never bothered updating.

When it comes to exchanging information and files no matter where, from whom or how, please be aware that there are certain dangers attached to it; realise the possible dangers of your actions and your naivety, and act accordingly.

4.10.18 Downloading

Downloading any data from unknown and untrustworthy sources while using company systems and networks could have a devastating effect on the business processes; you could face a situation of having your data lost, corrupted, or, in certain cases, modified. You should therefore aim to educate staff on the procedures of downloading information in a safe manner; this consists of ensuring downloadingf files only when it is absolutely necessary, scanning of the downloaded files with the corporate Anti-Virus/Anti-Trojan solution before opening it, etc.

For your convenience we have created a summarised "Internet Use Best Practices" section below; again, far from being an exhaustive list, it is aimed at giving you some basic pointers on safe Internet use.

4.10.19 Internet use best practices

- Java and ActiveX are blocked by default. Scripts containing Java and ActiveX pose a great danger due to their insecure nature, and the resulting problems could have devastating effects on your computer, not to mention the company. Please do not block, stop or tamper with any measure (i.e. group policy) that is in place to filter out these and if you are having problems purchasing an item or visiting a trusted web site, contact the IT department, Help Desk or the ISO office for assistance;
- Do not visit inappropriate web sites with objectionable content; pornography, gambling, warez (pirated software), hacker/hacking sites, as well as those generally considered as prohibited by your security policy;
- If the use of Instant Messaging (IM) applications is allowed, do not accept any attachments no matter of the file type, extension, or originator;

- Downloading software, files or anything else is prohibited. If you need any applications for your dayto- day business, contact either the IT department, the Help Desk or the ISO office. You will more than likely need to hand in a (software) request form signed by your manager to complete the process. If you do get clearance to download a piece of software, remember to never execute it before scanning them with the corporate Anti-Virus/Anti-Trojan software;
- All internet activity should continuously be monitored and the users should be aware that they could be held liable for visiting prohibited web sites, downloading illegal files and content, as well as face a penalty of having their access to the Internet limited (until they can prove that they are fully aware of the risks created by their actions).

chapter 5 vulnerabilities

The **S**.

5.1 paragraph 1

One

5.2 par 2

One iteration

chapter 6 threats

6.1 overview

A computer-based system has three separate but valuable components: hardware, software, and data.

Each of these assets offers value to different members of the community affected by the system.

A threat to a computing system is a set of circumstances that has the potential to cause loss or harm.

There are two types of network threats:

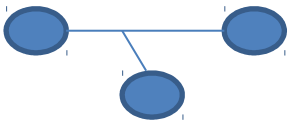
- **Logic attacks** – are known to exploit existing software bugs and vulnerabilities with the intent of crashing a system
- **Resource attacks** – are intended to overwhelm critical system resources such as CPU and RAM

There are many threats to a computer system, including human-initiated and computer-initiated ones. We have all experienced the results of inadvertent human errors, hardware design flaws, and software failures. But natural disasters are threats, too; they can bring a system down when the computer room is flooded or the data center collapses from an earthquake, for example.

A human who exploits vulnerability perpetrates an attack on the system. An attack can also be launched by another system, as when one system sends an overwhelming set of messages to another, virtually shutting down the second system's ability to function. We can say that a threat is blocked by control of vulnerability.

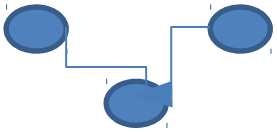
There are four types of security threats:

- An **interception** means that some unauthorized party has gained access to an asset. The outside party can be a person, a program, or a computing system. Examples of this type of failure are illicit copying of program or data files, or wiretapping to obtain data in a network. Although a loss may be discovered fairly quickly, a silent interceptor may leave no traces by which the interception can be readily detected.

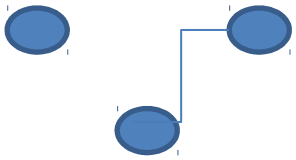


- An **interruption** means that an asset of the system becomes lost, unavailable, or unusable. An example is malicious destruction of a hardware device, erasure of a program or data file, or malfunction of an operating system file manager so that it cannot find a particular disk file.

- If an unauthorized party not only accesses but tampers with an asset, the threat is a **modification**. For example, someone might change the values in a database, alter a program so that it performs an additional computation, or modify data being transmitted electronically. It is even possible to modify hardware. Some cases of modification can be detected with simple measures, but other, more subtle, changes may be almost impossible to detect.



- Finally, an unauthorized party might create a **fabrication** of counterfeit objects on a computing system. The intruder may insert spurious transactions to a network communication system or add records to an existing database. Sometimes these additions can be detected as forgeries, but if skillfully done, they are virtually indistinguishable from the real thing.



A malicious attacker must have three things (MOM acronym):

- **method**: the skills, knowledge, tools, and other things with which to be able to pull off the attack
- **opportunity**: the time and access to accomplish the attack
- **motive**: a reason to want to perform this attack against this system

With an increasing amount of people getting connected to network, the security threats that cause harm are increasing also. Network security is a major part of a network that needs to be maintained because information is being passed between computers and is vulnerable to attack.

The biggest network threats are the following:

- viruses and worms
- trojan horses
- spam
- phishing
- packet sniffers
- spyware
- rootkits
- backdoors
- password attacks
- zombie computers and botnets

6.2 viruses and worms

A **computer virus** is a malware program that, when executed,replicates by inserting copies of itself (possibly modified) into other computer programs, data files, or the boot sectors of the hard drive; when this replication succeeds, the affected areas are then said to be "infected". Viruses often perform some type of harmful activity on infected hosts, such as stealing hard disk space or CPU time, accessing private information, corrupting data, displaying political or humorous messages on the user's screen, spamming their contacts, or logging their keystrokes.

A **computer worm** is a standalone malware computer program that replicates itself in order to spread to other computers. Often, it uses a computer network to spread itself, relying on security failures on the target computer to access it. Unlike a computer virus, it does not need to attach itself to an existing program.Worms almost always cause at least some harm to the network, even if only by consuming bandwidth, whereas viruses almost always corrupt or modify files on a targeted computer.

6.3 trojans

Similar to the mythical wooden horse used by the Greeks to invade Troy, a **Trojan Horse** is “a program in which malicious or harmful code is contained inside apparently harmless programming or data in such a way that it can get control and do its chosen form of damage, such as ruining the file allocation table on your hard disk”. On network, they are even more dangerous. They do not have the ability to self-replicate but to deliver destructive payloads and unload viruses, worms or spyware.

A Trojan often acts as a backdoor, contacting a controller which can then have unauthorized access to the affected computer. While Trojans and backdoors are not easily detectable by themselves, computers may appear to run slower due to heavy processor or network usage. Malicious programs are classified as Trojans if they do not attempt to inject themselves into other files (computer virus) or otherwise propagate themselves (worm).

6.4 spam

SPAM is “flooding the Internet with many copies of the same message, in an attempt to force the message on people who would not otherwise choose to receive it.”

Clicking on links in spam email may send users to phishing web sites or sites that are hosting malware. Spam email may also include malware as scripts or other executable file attachments. Definitions of spam usually include the aspects that email is unsolicited and sent in bulk.

The solution against them is spam filters which come with most of the email clients.

6.5 phishing

Phishing is an email fraud method in which the perpetrator sends out legitimate-looking email in an attempt to gather personal and financial information from recipients. Is one of the worst security threat over a network because a lot of people are vulnerable to giving out information that could cause money theft or identity theft.

Communications purporting to be from popular social web sites, auction sites, banks, online payment processors or IT administrators are commonly used to lure unsuspecting public. Phishing emails may contain links to websites that are infected with malware. Phishing is typically carried out by email spoofing or instant messaging, and it often directs users to enter details at a fake website whose look and feel are almost identical to the legitimate one. Phishing is an example of social engineering techniques used to deceive users, and exploits the poor usability of current web security technologies. Attempts to deal with the growing number of reported phishing incidents include legislation, user training, public awareness, and technical security measures. Many websites have now created secondary tools for applications, like maps for games, but they should be clearly marked as to who wrote them, and users should not use the same passwords anywhere on the internet.

Phishing is a continual threat that keeps growing to this day. The risk grows even larger in social media such as Facebook, Twitter, Myspace etc. Hackers commonly use these sites to attack persons using these media sites in their workplace, homes, or public in order to take personal and security information that can affect the user and the company (if in a workplace environment). Phishing is used to portray trust in the user since the user may not be able to tell that the site being visited or program being used is not real, and when this occurs is when the hacker has the chance to access the personal information such as passwords, usernames, security codes, and credit card numbers among other things.

There are filters designed to prevent this kind of threats, similar to spam filters.

6.6 packet sniffers

A **packet sniffer** is a device or program which allows eavesdropping on traffic traveling between networked computers; it will capture data that is addressed to other machines, saving it later for analysis.

As data streams flow across the network, the sniffer captures each packet and, if needed, decodes the packet's raw data, showing the values of various fields in the packet, and analyzes its content according to

the appropriate RFC or other specifications.

So again, personal information is at risk and the solution is to encrypt the data.

6.7 spyware

Spyware is software that aims to gather information about a person or organization without their knowledge and that may send such information to another entity without the consumer's consent, or that asserts control over a computer without the consumer's knowledge.

"Spyware" is mostly classified into four types: system monitors, trojans, adware, and tracking cookies. Spyware is mostly used for the purposes of tracking and storing Internet users' movements on the Web and serving up pop-up ads to Internet users.

Whenever spyware is used for malicious purposes, its presence is typically hidden from the user and can be difficult to detect. Some spyware, such as keyloggers, may be installed by the owner of a shared, corporate, or public computer intentionally in order to monitor users.

It is a ‘sneaky’ program that tracks and reports your computing activity without consent, such as browsing patterns in the more benign case or credit card numbers in more serious ones. It usually comes bundled with free software and automatically installs itself with the program you intended to use.

6.8 rootkits

A **rootkit** is a stealthy type of software, typically malicious, designed to hide the existence of certain processes or programs from normal methods of detection and enable continued privileged access to a computer. The term *rootkit* is a concatenation of "roo" (the traditional name of the privileged account on Unix operating systems) and the word "kit" (which refers to the software components that implement the tool). The term "rootkit" has negative connotations through its association with malware.

Rootkit installation can be automated, or an attacker can install it once they've obtained root or Administrator access. Obtaining this access is a result of direct attack on a system (i.e.), exploiting a known vulnerability (such as privilege escalation) or a password (obtained by cracking or social engineering). Once installed, it becomes possible to hide the intrusion as well as to maintain privileged access. The key is the root or Administrator access. Full control over a system means that existing software can be modified, including software that might otherwise be used to detect or circumvent it.

6.9 backdoors

A **backdoor** in a computer system is a method of bypassing normal authentication, securing unauthorized remote access to a computer, obtaining access to plaintext, and so on, while attempting to remain undetected. The backdoor may take the form of a hidden part of a program; a separate program may subvert the system through a rootkit.

Default passwords can function as backdoors if they are not changed by the user. Some debugging features can also act as backdoors if they are not removed in the release version

6.10 password attacks

Password attacks are attacks by hackers that are able to determine passwords or find passwords to different protected electronic areas. Many systems on a network are password protected and hence it would be easy for a hacker to hack into the systems and steal data. Password cracking is the process of recovering passwords from data that have been stored in or transmitted by a computer system. A common approach (brute-force attack) is to try guesses repeatedly for the password and check them against an available cryptographic hash of the password. There is no solution for to moment to prevent, just to create long and complicated password by using uppercase letters, special characters and numbers.

6.11 zombie computers and botnets

A z**ombie computer** is a computer that has been secretly compromised by hacking tools which allows a third party application to control the computer and its resources remotely

A **botnet** is a number of Internet computers that, although their owners are unaware of it, have been set up to forward transmissions to other computers on the internet. This is a major security threat, because the network can act as a hub that forwards malicious files let's say to other computers.

6.12 threats protection

We seek to protect hardware, software, and data; to make it particularly hard for an intruder to find data useful we scramble the data so that interpretation is meaningless without the intruder knows how the scrambling was done.

Encryption is the formal name for the scrambling process. We take data in their normal, unscrambled state, called cleartext, and transform them so that they are unintelligible to the outside observer; the transformed data are called enciphered text or ciphertext. Using encryption, security professionals can virtually nullify the value of an interception and the possibility of effective modification or fabrication.. Encryption clearly addresses the need for confidentiality of data. Additionally, it can be used to ensure integrity; data that cannot be read generally cannot easily be changed in a meaningful manner. Encryption is the basis of protocols that enable us to provide security while accomplishing an important system or network task. A protocol is an agreed-on sequence of actions that leads to a desired result. For example, some operating system protocols ensure availability of resources as different tasks and users request them. Thus, encryption can also be thought of as supporting availability. That is, encryption is at the heart of methods for ensuring all aspects of computer security. Although encryption is an important tool in any computer security tool kit, we should not overrate its importance. Encryption does not solve all computer security problems, and other tools must complement its use. Furthermore, if encryption is not used properly, it may have no effect on security or could even degrade the performance of the entire system. Weak encryption can actually be worse than no encryption at all, because it gives users an unwarranted sense of protection. Therefore, we must understand those situations in which encryption is most useful as well as ways to use it effectively

Another solution is to use a firewall; a **firewall** is a network security system that controls the incoming and outgoing network traffic based on an applied rule set. A firewall establishes a barrier between a trusted, secure internal network and another network (e.g., the Internet) that is assumed not to be secure and trusted. Firewalls exist both as software to run on general purpose hardware and as a hardware appliance. Many hardware-based firewalls also offer other functionality to the internal network they protect, such as acting as a DHCP server for that network.

chapter 7 passive attacks

There are two main types of passive attack:

- traffic analysis
- Non-evasive eavesdropping and monitoring of transmissions

7.1 traffic analysis

7.1.1 Military roots

Traffic analysis is a key part of signal intelligence and electronic warfare. Michael Hermann, who has served as chair of the UK Joint Intelligence Committee, in his book 'Intelligence Power in Peace and War' describes the value of extracting data from non-textual (to be understood as 'not content') sources: These non-textual techniques can establish targets' locations, order of-battle and movement. Even when messages are not being deciphered, traffic analysis of the target's C3I system and its patterns of behavior provides indications of his intentions and states of mind, in rather the same way as a neurologist develops insights about a silent patient by studying EEG traces from the brain. Traffic analysis was used in military circles even before the invention of wireless communications. Anderson in his book [3] mentions that in the trench warfare of World War I, the earth returns of the telegraph communication of the enemy was used to extract information up to a few hundred yards away from the transmitting station. Traffic analysis though became an extremely potent source of intelligence when wireless communication became popular, particularly in naval and air operations. Ships at see had to balance the value of communicating against the threat of being detected via direction finding if they transmit. When transmitting strict standards, governing call-signs and communication, had to be adhered too in order to minimize the information that traffic analysis could provide. Another example of traffic analysis providing valuable intelligence (by Herman) is the reconstruction of the structure of the network structure of the German Air Force radio in 1941 by the British, confirming that a unit was composed of nine and not twelve planes. This allowed a more accurate estimate of the total strength of their opponent. Identification of radio equipment can also be used to detect accurate movements of units: each transmitter has characteristics such as the unintentional frequency modulations, the shape of the transmitter turn-on signal transient, the precise center of frequency modulation, etc.

7.2 eavesdropping and transmission monitoring

One iteration

chapter 8 active attacks

The **S**.

8.1 paragraph 1

One

8.2 par 2

One iteration

chapter 9 hash functions

9.1 overview

Hash functions take a message of arbitrary length as input and generate a fixed length digest (checksum). The length of the digest depends on the function used, but in general is between 128 and 512 bits.

The hash functions are used in 3 main areas:

- assure the integrity of a message (or of a downloaded file) by attaching the generated digest to the message itself. The receiver recomputes the digest using the received message and compares it against the digest generated by the sender.
- are part of the creation of the digital signature
- password storage – password are (almost) never stored in their original form. What is stored, in general, is a hash of the password. When a user introduces a password, its hash is computed and is compared with the stored hash.

The most used hash functions are those in the MD and the SHA families – namely MD5 and SHA-1 and the newest ones SHA-2 and SHA-3. Another hash function of interest is RipeMD-160. The MD functions generate a 128 bit digest and were designed by the company RSA Security. While MD5 is still widespread, MD4 has been broken and is deemed insecure. SHA-1 and RipeMD-160 are considered safe for now. While SHA-2 is an extension of SHA-1, SHA_3 features a brand new algorithm for computing the hash.

Starting with the newest function, here is a list of hash functions of practical interest.

- SHA-3 uses the Keccak algorithm, a sponge construction in which message blocks are XORed into a subset of the state, which is then transformed as a whole. In the version used in SHA-3, the state consists of a 5×5 array of 64-bit words, 1600 bits total. The standardization process is not finished yet as of April 2015.
- SHA-2 includes significant changes from its predecessor, SHA-1. The SHA-2 family consists of six hash functions with digests (hash values) that are 224, 256, 384 or 512 bits: **SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224, SHA-512/256**.
- SHA-1 – Secure Hash Algorithm. Published by the US Government. Its specification is the object of FIPS 180-1 (April 1995). FIPS stands for Federal Information Processing Standards. Produces a 160 bit digest (5 32-bit words).
- RipeMD-160 – designed as a replacement for the MD series. It produces a digest of 160 bits (or 20 bytes, if you want).
- MD5 – Message Digest Algorithm 5. Developed by RSA Labs. Produces a 128 bit digest. Still in use, especially for message (download) integrity check.
- MD2, MD4 – Older hash algorithms from RSA Data Security. Since they have known flaws, they are only of historic interest.

9.2 characteristics

A good hash function is supposed to exhibit the following characteristics:

9.3 MD5

9.3.1 General description

One iteration

9.3.2 history

One iteration

9.3.3 algorithm description

One iteration

9.3.4 some examples

One iteration

9.3.5 cryptanalysis

One iteration

9.4 SHA-1

9.4.1 General description

One iteration

9.4.2 history

One iteration

9.4.3 algorithm description

One iteration

9.4.4 some examples

One iteration

9.4.5 cryptanalysis

One iteration

9.5 SHA-2

One iteration

9.6 SHA-3

9.6.1 general description

One iteration

9.6.2 history

One iteration

9.6.3 algorithm description

One iteration

9.6.4 some examples

One iteration

9.6.5 cryptanalysis

One iteration

9.7 practical collisions

One iteration

chapter 10 encryption systems

10.1 what is encryption?

Encryption is the conversion of electronic data into another form, called ciphertext, which cannot be easily understood by anyone except authorized parties.

Computer encryption is based on the science of cryptography, which has been used as long as humans have wanted to keep information secret. Before the digital age, the biggest users of cryptography were governments, particularly for military purposes.

The Greek historian Plutarch wrote, for example, about Spartan generals who sent and received sensitive messages using a scytale, a thin cylinder made out of wood. The general would wrap a piece of parchment around the scytale and write his message along its length. When someone removed the paper from the cylinder, the writing appeared to be a jumble of nonsense. But if the other general receiving the parchment had a scytale of similar size, he could wrap the paper around it and easily read the intended message.

The Greeks were also the first to use ciphers, specific codes that involve substitutions or transpositions of letters and numbers.

As long as both generals had the correct cipher, they could decode any message the other sent. To make the message more difficult to decipher, they could arrange the letters inside the grid in any combination.

Most forms of cryptography in use these days rely on computers, simply because a human-based code is too easy for a computer to crack. Ciphers are also better known today as algorithms, which are the guides for encryption -- they provide a way in which to craft a message and give a certain range of possible combinations. A key, on the other hand, helps a person or computer figure out the one possibility on a given occasion.

Computer encryption systems generally belong in one of two categories:

1. Symmetric-key encryption
2. Public-key encryption

10.2 types of encryption

10.2.1 symmetric key encryption

In symmetric-key schemes, the encryption and decryption keys are the same. Thus communicating parties must have the same key before they can achieve secret communication.

10.2.2 public key encryption

In public-key encryption schemes, the encryption key is published for anyone to use and encrypt messages. However, only the receiving party has access to the decryption key that enables messages to be read. Public-key encryption was first described in a secret document in 1973; before then all encryption schemes were symmetric-key (also called private-key).

A publicly available public key encryption application called Pretty Good Privacy (PGP) was written in 1991 by Phil Zimmermann, and distributed free of charge with source code; it was purchased by Symantec in 2010 and is regularly updated.

10.3 DES – Data Encryption Standard

The Data Encryption Standard (DES) is an outdated symmetric-key method of data encryption.

DES works by using the same key to encrypt and decrypt a message, so both the sender and the receiver must know and use the same private key. Once the go-to, symmetric-key algorithm for the encryption of electronic data, DES has been superseded by the more secure Advanced Encryption Standard (AES) algorithm.

10.3.1 general description

One iteration

10.3.2 history

Originally designed by researchers at IBM in the early 1970s, DES was adopted by the U.S. government as an official Federal Information Processing Standard (FIPS) in 1977 for the encryption of commercial and sensitive yet unclassified government computer data. It was the first encryption algorithm approved by the U.S. government for public disclosure. This ensured that DES was quickly adopted by industries such as financial services, where the need for strong encryption is high. The simplicity of DES also saw it used in a wide variety of embedded systems, smart cards, SIM cards and network devices requiring encryption like modems, set-top boxes and routers.

10.3.3 algorithm description

One iteration

10.3.4 some examples

One iteration

10.3.5 cryptanalysis

One iteration

10.4 AES – Advanced Encryption Standard

The Advanced Encryption Standard or AES is a symmetric block cipher used by the U.S. government to protect classified information and is implemented in software and hardware throughout the world to encrypt sensitive data.

10.4.1 general description

One iteration

10.4.2 history

The origins of AES date back to 1997 when the National Institute of Standards and Technology (NIST) announced that it needed a successor to the aging Data Encryption Standard (DES) which was becoming vulnerable to brute-force attacks.

10.4.3 algorithm description

One iteration

10.4.4 some examples

One iteration

10.4.5 cryptanalysis

One iteration

10.5 the Diffie-Hellman key exchange algorithm

Diffie-Hellman key exchange, also called exponential key exchange, is a method of digital encryption that uses a number raised to specific powers to produce decryption keys that are never directly transmitted, making the task of a would-be code breaker mathematically overwhelming.

To implement Diffie-Hellman, the two end users Alice and Bob, while communicating over a channel they know to be private, mutually agree on positive whole numbers p and q, such that p is a prime number and q is a generator of p. The generator q is a number that, when raised to positive whole-number powers less than p, never produces the same result for any two such whole numbers. The value of p may be large but the value of q is usually small.

Once Alice and Bob have agreed on p and q in private, they choose positive whole-number personal keys a and b, both less than the prime-number modulus p. Neither user divulges their personal key to anyone; ideally they memorize these numbers and do not write them down or store them anywhere. Next, Alice and Bob compute public keys a* and b* based on their personal keys according to the formulas

a* = q^a mod p

and

b* = q^b mod p

The two users can share their public keys a* and b* over a communications medium assumed to be insecure, such as the Internet or a corporate wide area network (WAN). From these public keys, a number x can be generated by either user on the basis of their own personal keys. Alice computes x using the formula

10.6 RSA

RSA is a cryptosystem for public-key encryption, and is widely used for securing sensitive data, particularly when being sent over an insecure network such as the Internet.

RSA was first described in 1977 by Ron Rivest, Adi Shamir and Leonard Adleman of the Massachusetts Institute of Technology. Public-key cryptography, also known as asymmetric cryptography, uses two different but mathematically linked keys, one public and one private. The public key can be shared with everyone, whereas the private key must be kept secret. In RSA cryptography, both the public and the private keys can encrypt a message; the opposite key from the one used to encrypt a message is used to decrypt it. This attribute is one reason why RSA has become the most widely used asymmetric algorithm: It provides a method of assuring the confidentiality, integrity, authenticity and non-reputability of electronic communications and data storage.

Many protocols like SSH, OpenPGP, S/MIME, and SSL/TLS rely on RSA for encryption and digital signature functions. It is also used in software programs -- browsers are an obvious example, which need to establish a secure connection over an insecure network like the Internet or validate a digital signature. RSA signature verification is one of the most commonly performed operations in IT.

10.7 the encryption nowadays

Few organizations today have access to truly private and secure networks; instead, they share network infrastructure with other organizations. As a result, information traveling over these public or virtual private

networks is often vulnerable to interception. Quite rightly, many of today's data privacy requirements and standards include, as a baseline level of protection, a mandate to protect data in motion. While organizations can choose to encrypt selected data at the application level or within databases or other storage environments, the bulk protection of data flowing over a network provides a blunt but very effective instrument for adding an extra layer of security. Network encryption guards against regulated data inadvertently being sent in the clear and also provides valuable protection for all other classes of data that perhaps do not justify dedicated protection but nonetheless are still considered sensitive. Although network-level encryption is a relatively mature technology, organizations need to make several choices when deciding what kind of network encryption to deploy

Standalone network encryption platforms are particularly valuable for high-speed connections between data centers. Globally interconnected organizations and service providers require the combination of optimized bandwidth, unshakeable resilience, and security for critical systems such as storage area networks (SANs), transaction systems, and cloud computing. The ability to secure these backbone connections as transparently as possible becomes a critical success factor for enterprises and a valuable differentiator for network service providers.

10.7.1 which networks should have their traffic encrypted?

Most networks are 'open' to some degree, but some are much more open than others. Internal wired networks might be considered vulnerable only for the most sensitive data, since they still suffer from the threat of insider attacks, whereas backbone networks and wide area network (WAN) connections usually deserve more consideration as they typically use shared pipes from external service providers. In almost all settings, organizations will want to encrypt traffic over wireless local area networks (LANs), wireless WANs, and, of course, the Internet. This page focuses mainly on WAN encryption.

10.7.2 should traffic be encrypted at Layer 2 or Layer 3 in the OSI Network Model?

At stake in this choice are overhead and the potential waste of bandwidth. Applying encryption at Layer 3, using well-known protocols such as IPsec, creates the need to preserve routing information used by equipment throughout the network. This imposes a significant overhead, ultimately affecting capacity and latency. Layer 2 encryption operates at a lower layer and is independent of the routing information and flow-management techniques that exist at Layer 3, and is more efficient in most cases. That said, IPsec, remains the most common form of network encryption for all but high-speed data-center-to-data-center connections where bandwidth and latency are most critical.

10.7.3 are your security needs best served by embedded or standalone encryption?

Since network-level encryption is a relatively mature technology, it is commonly available as an embedded or native feature of routing or switching equipment. Standalone encryption platforms provide an alternative to embedded encryption—one that delivers a higher level of assurance and benefits from purpose-built key management capabilities. Standalone encryption platforms are independently certified against security benchmarks such as FIPS 140 and Common Criteria, offer tamper resistance, and offer features that enable organizations to enforce a strong separation of duties between network administrators and security officers.

10.8 risks associated with network security

Attackers can "eavesdrop" on unencrypted data traveling over a network, not only impacting privacy but potentially opening the potential to modify or substitute data as a way to stage more sophisticated attacks.

Because industry mandates often require protection for data in motion, organizations that do not implement this protection risk fines, embarrassing data breach disclosure statements, and resulting damage to their reputation.

Depending on the application, encryption capabilities embedded in routers and switches may not offer the combination of security and performance you need.

10.9 example of one country which uses encryption systems

The National Security Agency (NSA) is an intelligence organization of the United States government, responsible for global monitoring, collection, and processing of information and data for foreign intelligence and counterintelligence purposes - a discipline known as Signals intelligence (SIGINT). NSA is concurrently charged with protection of U.S. government communications and information systems against penetration and network warfare.

The large number of encryption systems that NSA has developed can be grouped by application.

10.9.1 record traffic encryption

During World War II, written messages (known as record traffic) were encrypted off line on special, and highly secret, rotor machines and then transmitted in five letter code groups using Morse code or teletypewriter circuits, to be decrypted off-line by similar machines at the other end. The SIGABA rotor machine, developed during this era continued to be used until the mid-1950s, when it was replaced by the KL-7, which had more rotors.

The KW-26 ROMULUS was a second generation encryption system in wide use that could be inserted into teletypewriter circuits so traffic was encrypted and decrypted automatically. It used electronic shift registers instead of rotors and became very popular (for a COMSEC device of its era), with over 14,000 units produced. It was replaced in the 1980s by the more compact KG-84, which in turn was superseded by the KG-84-interoperable KIV-7.

10.9.2 fleet broadcast

U.S. Navy ships traditionally avoid using their radios to prevent adversaries from locating them by direction finding. The Navy also needs to maintain traffic security, so it has radio stations constantly broadcasting a stream of coded messages. During and after World War II, Navy ships copied these fleet broadcasts and used specialized call sign encryption devices to figure out which messages were intended for them. The messages would then be decoded off line using SIGABA or KL-7 equipment.

The second generation KW-37 automated monitoring of the fleet broadcast by connecting in line between the radio receiver and a teleprinter. It, in turn, was replaced by the more compact and reliable third generation KW-46.

10.9.3 internet

NSA has approved a variety of devices for securing Internet Protocol communications. These have been used to secure the Secret Internet Protocol Router Network (SIPRNet), among other uses.

The first commercial network layer encryption device was the Motorola Network Encryption System (NES). The system used the SP3 and KMP protocols defined by the NSA Secure Data Network System (SDNS) and were the direct precursors to IPsec. The NES was built in a three part architecture that used a small cryptographic security kernel to separate the trusted and untrusted network protocol stacks.

The SDNS program defined a Message Security Protocol (MSP) that was built on the use X.509 defined certificates. The first NSA hardware built for this application was the BBN Safekeeper. The Message Security Protocol was a precursor to the IETF Privacy Enhance Mail (PEM) protocol. The BBN Safekeeper provided a high degree of tamper resistance and was one of the first devices used by commercial PKI companies.
iteration

10.10 some examples of the encryption on the internet

10.10.1 HTTPS

HTTPS is a communications protocol for secure communication over a computer network, with especially wide deployment on the Internet. Technically, it is not a protocol in and of itself; rather, it is the result of simply

layering the Hypertext Transfer Protocol (HTTP) on top of the SSL or TLS protocol, thus adding the security capabilities of SSL/TLS to standard HTTP communications. The main motivation for HTTPS is to provide authentication of the visited website and to protect the privacy and integrity of exchanged data.The security of HTTPS is therefore that of the underlying TLS, which uses long-term public and secret keys to exchange a short term session key to encrypt the data flow between client and server. X.509 certificates are used to guarantee one is talking to the partner with whom one wants to talk. As a consequence, certificate authorities and a public key infrastructure are necessary to verify the relation between the owner of a certificate and the certificate, as well as to generate, sign, and administer the validity of certificates. While this can be more beneficial than verifying the identities via a web of trust, the 2013 mass surveillance disclosures made it more widely known that certificate authorities are a weak point from a security standpoint, allowing man-in-the-middle attacks. Another important property in this context is perfect forward secrecy (PFS), so the short-term session key cannot be derived from the long-term asymmetric secret key; however, PFS is not widely adopted.

10.10.2 TLS and SSL

Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), are cryptographic protocols designed to provide communications security over a computer network.

SSL is the secure communications protocol of choice for a large part of the Internet community. There are many applications of SSL in existence, since it is capable of securing any transmission over TCP. Secure HTTP, or HTTPS, is a familiar application of SSL in e-commerce or password transactions. According to the Internet Draft of the SSL Protocol, the point of the protocol “is to provide privacy and reliability between two communicating applications.”

The protocol release further explains that three points combine to provide connection security. These points are:

- Privacy - connection through encryption
- Identity authentication – identification through certificates
- Reliability –dependable maintenance of a secure connection through message integrity checking.

10.10.3 PGP

Pretty Good Privacy (PGP) is a data encryption and decryption computer program that provides cryptographic privacy and authentication for data communication. PGP is often used for signing, encrypting, and decrypting texts, e-mails, files, directories, and whole disk partitions and to increase the security of e-mail communications. It was created by Phil Zimmermann in 1991.

PGP and similar software follow the OpenPGP standard (RFC 4880) for encrypting and decrypting data.

The program PGP it’s a free software for non-commercial use and accessible in www.pgpi.com. It’s the most used system all over the world by particular users and also big companies. The newest versions of the software are very easy to use and communicates with the most used email softwares (Outlook, Netscape Mail, Eudora, etc.).

10.10.4 SET

Secure Electronic Transaction (SET) was a communications protocol standard for securing credit card transactions over insecure networks, specifically, the Internet. SET was not itself a payment system, but rather a set of security protocols and formats that enabled users to employ the existing credit card payment infrastructure on an open network in a secure fashion. However, it failed to gain attraction in the market. VISA now promotes the 3-D Secure scheme.ne iteration

chapter 11 the public key infrastructure

The **S**.

11.1 paragraph 1

One

11.2 par 2

One iteration

chapter 12 the digital signature standard

12.1 what is a digital signature?

Digital signatures are essential in today's world to verify who the sender of a document is. A digital signature is represented in a computer as a string of binary digits. The signature is computer using a set of rules and parameters (algorithm) such that the identity of the person signing the document as well as the originality of the data can be verified. The signature is generated by the use of a private key. A private key is known only to the user. The signature is verified makes use of a public key which corresponds to the private key. With every user having a public/private key pair, this is an example of public-key cryptography. Public keys, which are known by everyone, can be used to verify the signature of a user. The private key, which is never shared, is used in signature generation, which can only be done by the user.

Digital signatures are used to detect unauthorized modifications to data. Also, the recipient of a digitally signed document in proving to a third party that the document was indeed signed by the person who it is claimed to be signed by. This is known as nonrepudiation, because the person who signed the document cannot repudiate the signature at a later time. Digital signature algorithms can be used in e-mails, electronic funds transfer, electronic data interchange, software distribution, data storage, and just about any application that would need to assure the integrity and originality of data.

12.2 what is DSS?

Digital Signature Standard (DSS) is the digital signature algorithm (DSA) developed by the U.S. National Security Agency (NSA) to generate a digital signature for the authentication of electronic documents. DSS was put forth by the National Institute of Standards and Technology (NIST) in 1994, and has become the United States government standard for authentication of electronic documents. DSS is specified in Federal Information Processing Standard (FIPS) 186.

This Standard specifies algorithms for applications requiring a digital signature, rather than a written signature. A digital signature is represented in a computer as a string of bits. A digital signature is computed using a set of rules and a set of parameters that allow the identity of the signatory and the integrity of the data to be verified. Digital signatures may be generated on both stored and transmitted data.
Signature generation uses a private key to generate a digital signature; signature verification uses a public key that corresponds to, but is not the same as, the private key. Each signatory possesses a private and public key pair. Public keys may be known by the public; private keys are kept secret. Anyone can verify the signature by employing the signatory's public key. Only the user that possesses the private key can perform signature generation. A hash function is used in the signature generation process to obtain a condensed version of the data to be signed; the condensed version of the data is often called a message digest. The message digest is input to the digital signature algorithm to generate the digital signature.

The hash functions to be used are specified in the Secure Hash Standard (SHS), FIPS 180. FIPS approved digital signature algorithms shall be used with an appropriate hash function that is specified in the SHS. The digital signature is provided to the intended verifier along with the signed data. The verifying entity verifies the signature by using the claimed signatory's public key and the same hash function that was used to generate the signature. Similar procedures may be used to generate and verify signatures for both stored and transmitted data.

12.3 the digital signature algorithm - DSA

DSA refers to a standard for digital signatures. It was introduced in 1991 by the National Institute of Standards and Technology (NIST) as a better method of creating digital signatures. Along with RSA, DSA is considered one of the most preferred digital signature algorithms used today.

Unlike DSA, most digital signature types are generated by signing message digests with the private key of the originator. This creates a digital thumbprint of the data. Since just the message digest is signed, the signature is generally much smaller compared to the data that was signed. As a result, digital signatures impose less load on processors at the time of signing execution, use small volumes of bandwidth, and generate small volumes of cipher text intended for cryptanalysis.

DSA, on the other hand, does not encrypt message digests using private key or decrypt message digests using public key. Instead, it uses unique mathematical functions to create a digital signature consisting of two 160-bit numbers, which are originated from the message digests and the private key. DSAs make use of the public key for authenticating the signature, but the authentication process is more complicated when compared with RSA.

The digital signature procedures for RSA and DSA are usually regarded as being equal in strength. Because DSAs are exclusively used for digital signatures and make no provisions for encrypting data, it is typically not subject to import or export restrictions, which are often enforced on RSA cryptography.

12.3.1 DSA Parameters

- **p** = a prime modulus, where $2^{L-1} < p < 2^L$ for $512 \le L \le 1024$ and L is a multiple of 64. So L will be one member of the set {512, 576, 640, 704, 768, 832, 896, 960, 1024}.
- **q** = a prime divisor of p-1, where $2^{159} < q < 2^{160}$
- **g** = $h^{(p-1)/q} \bmod p$, where h is any integer with $1 < h < p-1$ such that $h^{(p-1)/q} \bmod p > 1$ (g has order q mod p)
- **x** = a randomly integer with $0 < x < q$
- **y** = $g^x \bmod p$
- **k** = a randomly generated integer with $0 < k < q$

12.3.2 generation of primes p and q

Now we have to know how to generate p and q. It will be describe step by step:

The prime generation scheme starts by using the SHA and a user supplied SEED to construct a prime, q, in the range $2^{159} < q < 2^{160}$. Once this is accomplished, the same SEED value is used to construct an X in the range $2^{L-1} < X < 2^L$. The prime, p, is then formed by rounding X to a number congruent to 1 mod 2q as described below. An integer x in the range $0 \le x < 2$ g may be converted to a g-long sequence of bits by using its binary expansion as shown below:

$x = x_1 * 2^{g-1} + x_2 * 2^{g-2} + ... + x_{g-1} * 2 + x_g \rightarrow \{ x_1,..., x_g \}.$

Conversely, a g-long sequence of bits $\{ x_1,..., x_g \}$ is converted to an integer by the rule:

$\{ x_1,..., x_g \} \rightarrow x_1 * 2^{g-1} + x_2 * 2^{g-2} + ... + x_{g-1} * 2 + x_g.$

Note that the first bit of a sequence corresponds to the most significant bit of the corresponding integer and the last bit to the least significant bit.

12.3.3 steps

As specified in - http://www.ijettcs.org/Volume4Issue2/IJETTCS-2014-12-22-136.pdf

Let L -1 = n* 160 + b, where both b and n are integers and $0 \le b < 160$.

Step1. Choose an arbitrary sequence of at least 160 bits and call it SEED. Let g be the length of SEED in bits.

Step2. Compute U = SHA-1[SEED] XOR SHA-1[(SEED+ 1) mod 2 g].

Step3. Form q from U by setting the most significant bit (the 2 159 bit) and the least significant bit to 1. In terms of Boolean operations, q = U OR 2159 OR 1. Note that 2159 < q < 2160 .

Step4. Use a robust primality testing algorithm to test whether q is prime 1 .

Step5. If q is not prime, go to step 1.

Step6. Let counter = 0 and offset = 2. Step7.

For k = 0,..., n let Vk = SHA-1[(SEED + offset + k) mod 2g]. 1 A robust primality test is one where the probability of a non-prime number passing the test is at most 2-80

Step8. Let W be the integer W = V0 + V1* 2160 + ... + Vn-1* 2(n-1)* 160 + (Vn mod 2b) * 2 n* 160 and let X = W + 2L-1 . Note that 0 ≤ W < 2L-1 and hence 2L-1 ≤ X < 2L .

Step9. Let c = X mod 2q and set p = X -(c -1). Note that p is congruent to 1 mod 2q.

Step10. If p < 2L-1 , then go to step 13.

Step11. Perform a robust primality test on p.

Step12. If p passes the test performed in step 11, go to step 15.

Step13. Let counter = counter + 1 and offset = offset + n + 1.

Step14. If counter ≥ 212 = 4096 go to step 1, otherwise (i. e. if counter < 4096) go to step 7.

Step15. Save the value of SEED and the value of counter for use in certifying the proper generation of p and q.

- g = h(p-1)/ q mod p, where h is any integer with 1 < h < p -1 such that h(p-1)/ q mod p > 1 (g has order q mod p)
- x = a randomly or pseudorandomly generated integer with 0 < x < q
- y = gx mod p
- k = a randomly or pseudorandomly generated integer with 0 < k < q

The parameters p, q, and g are made public. The users will have the private key, x, and the public key y. The parameters x and k are used for signature generation and must be kept private and k will be randomly or pseudorandomly generated for each signature. This part seems to be straightforward so far. The signature of the message M will be a pair of the numbers r and s which will be computed from the following equations.

- r = (gk mod p) mod q
- s = (k-1 (SHA(M) + xr)) mod q

k -1 is the multiplicative inverse of k (mod q). The value of SHA(M) is a 160-bit string which is converted into an integer according to the SHS standard. Then the signature is sent to the verifier.

12.3.4 verification

Before getting the digitally signed message the receiver must know the parameters p, q, g, and the sender's public key y.

We will let M′, r′, s′ be the received versions of M, r, and s. To verify the signature the verifying program must check to see that 0 < r′ < q and 0 < s′ < q and if either fails the signature should be rejected. If both of the conditions are satisfied then we will compute

12.4 sensitivity

With DSA, the entropy, secrecy, and uniqueness of the random signature value k is critical. It is so critical that violating any one of those three requirements can reveal the entire private key to an attacker. Using the same value twice (even while keeping k secret), using a predictable value, or leaking even a few bits of k in each of several signatures, is enough to break DSA.

In December 2010, a group calling itself *fail0verflow* announced recovery of the ECDSA private key used by Sony to sign software for the PlayStation 3 game console. The attack was made possible because Sony failed to generate a new random k for each signature.

This issue can be prevented by deriving k deterministically from the private key and the message hash, as described by RFC 6979. This ensures that k is different for each H(m) and unpredictable for attackers who do not know the private key x.

chapter 13 authentication techniques

The Network/computer security hinges on two very simple goals:

- Keeping unauthorized persons from gaining access to resources
- Ensuring that authorized persons can access the resources they need

Access permissions works only if you are able to verify the identity of the user who is attempting to access the resources. That's where authentication comes in.

Authentication is the process of confirming the identification of a user (or in some cases, a machine) that is trying to log on or access resources.

It is easy to confuse authentication with another element of the security plan: authorization. While authentication verifies the user's identity, authorization verifies that the user in question has the correct permissions and rights to access the requested resource. As you can see, the two work together. Authentication occurs first, then authorization.

13.1 Authentication occurrences

- Logon authentication
  - It is required by most network operating systems.
  - Users need to authenticate in order to log onto the network.
- Network access authentication
  - Verifies the user's identity to each network service that the user attempts to access.
  - Authentication process is, in most cases, transparent to the user once he or she has logged on. Otherwise, the user would have to reenter the password or provide other credentials every time he or she wanted to access another network service or resource.
- IPSec authentication
  - Provides a means for users to encrypt and/or sign messages that are sent across the network to guarantee confidentiality, integrity, and authenticity.
  - An important consideration is that both the sending and receiving computers must be configured to use a common authentication method or they will not be able to engage in secured communications.
- Remote authentication
  - Remote users can be authenticated via a Remote Authentication Dial-In User Service (RADIUS) or the Internet Authentication Service (IAS).
- Single Sign-On (SSO)
  - Single Sign-On (SSO) is a feature that allows a user to use one password (or smart card) to authenticate to multiple servers on a network without reentering credentials.

Users don't have to remember multiple passwords or keep going through the authentication process over and over to access different resources.

13.2 Means to provide authentication credentials

- **Password authentication**
  - To log onto a computer or network, you enter a user account name and the password

assigned to that account. This password is checked against a database that contains all authorized users and their passwords.

  - To preserve the security of the network, passwords must be “strong,” that is, they should contain a combination of alpha and numeric characters and symbols, they should not be words that are found in a dictionary, and they should be relatively long (eight characters or more). In short, they should not be easily guessed.
  - Password authentication is vulnerable to a password “cracker” who uses a brute force attack (trying every possible combination until hitting upon the right one) or who uses a protocol “sniffer” to capture packets if passwords are not encrypted when they are sent over the network.

- **Smart card authentication**
  - Smart cards are credit card-sized devices that hold a small computer chip, which is used to store public and private keys and other personal information used to identify a person and authenticate him or her to the system.
  - Logging onto the network with a smart card requires that you physically insert the card into (or slide it through) a reader and then enter a Personal Identification Number (PIN) in much the same way that you use an ATM card to access an automatic teller machine.
  - Smart cards use cryptography-based authentication and provide stronger security than a password because in order to gain access, the user must be in physical possession of the card and must know the PIN.
- **Biometric authentication**
  - An even more secure type of authentication than smart cards

Biometric authentication involves the use of biological statistics that show that the probability of two people having identical biological characteristics such as fingerprints is infinitesimally small; thus, these biological traits can be used to positively identify a person.

13.3 Protocols used for authentication

An authentication protocol is a type of cryptographic protocol with the purpose of authenticating entities wishing to communicate securely.

There are different authentication protocols such as:

- Kerberos
- Microsoft CHAP
- Secure Socket Layer (SSL)
- Host Identity Protocol (HIP)
- Microsoft Network LAN Manager (NTLM)
- Password Authentication Protocol (PAP)
- Extensible Authentication Protocol (EAP)
- Shiva Password Authentication Protocol (SPAP)
- Protected Extensible Authentication Protocol (PEAP)
- Remote Authentication Dial-In User Service (RADIUS)
- Challenge-Handshake Authentication Protocol (CHAP)

chapter 14 kerberos

14.1 Overview

Kerberos is a secure method for authenticating a request for a service in a computer network. It was developed at MIT and the name is taken from Greek mythology: Kerberos was a three-headed dog who guarded the gates of Hades. The three heads of Kerberos are:

1. Key Distribution Center (KDC)
2. Client user
3. Server with the desired service to access

The KDC is installed as part of the domain controller and performs two service functions: the Authentication Service (AS) and the Ticket-Granting Service (TGS). As exemplified in Figure 1, three exchanges are involved when the client initially accesses a server resource:

1. AS Exchange
2. TGS Exchange
3. Client/Server (CS) Exchange

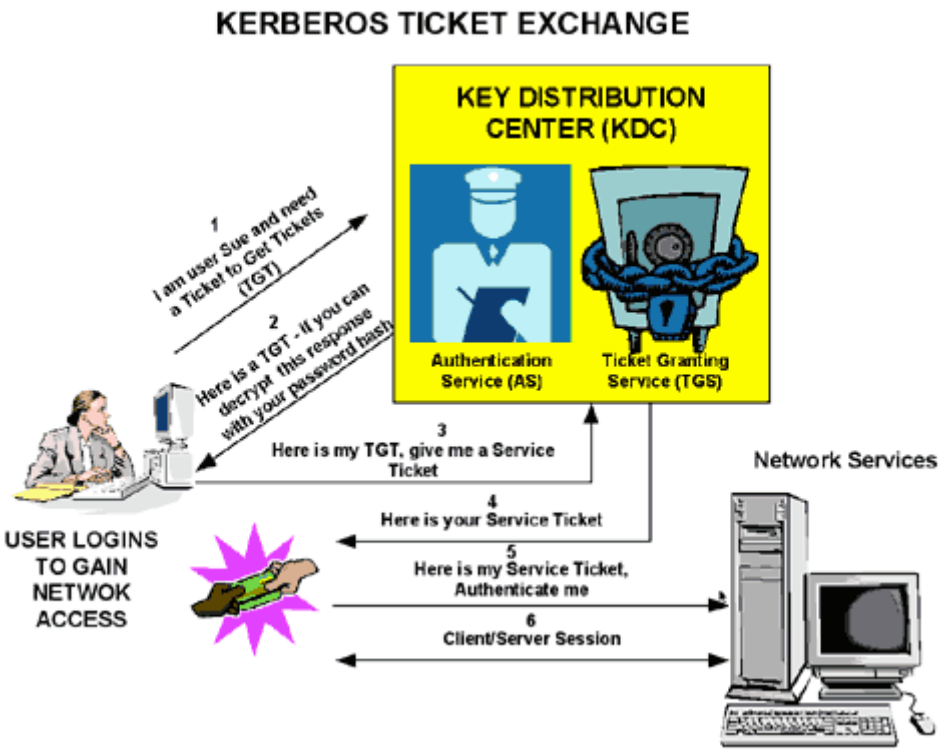


Figure 14.1

A Kerberos realm defines what Kerberos manages in terms of who can access what. This realm holds the client, the service or host wanted to request and the KDC. It is created by the admin and encompasses all that is available to access.

14.2 Authentication Service (AS) Exchange

When initially logging on to a network, users must negotiate access by providing a **log-in name** and **password** in order to be verified by the AS portion of a KDC within their domain. The KDC has access to Active Directory user account information.

Once successfully authenticated, the user is granted a **Ticket to Get Tickets (TGT)** that is valid for the local domain. The TGT has a default lifetime of 10 hours and may be renewed throughout the user's log-on session without requiring the user to re-enter his password. The TGT is cached on the local machine in volatile memory space and used to request sessions with services throughout the network.

If the KDC approves the client's request for a TGT, the reply (referred to as the AS reply) will include two sections:

- a TGT encrypted with a key that only the KDC (TGS) can decrypt
- a session key encrypted with the user's password hash to handle future communications with the KDC.

Because the client system cannot read the TGT contents, it must blindly present the ticket to the TGS for service tickets. The TGT includes:

- time to live parameters
- authorization data
- a session key to use when communicating with the client
- the client's name.

14.3 TGS Exchange

The user presents the TGT to the TGS portion of the KDC when desiring access to a server service. The TGS on the KDC authenticates the user's TGT and creates a ticket and session key for both the client and the remote server. This information, known as the service ticket, is then cached locally on the client machine.

The TGS receives the client's TGT and reads it using its own key. If the TGS approves of the client's request, a service ticket is generated for both the client and the target server. The client reads its portion using the TGS session key retrieved earlier from the AS reply. The client presents the server portion of the TGS reply to the target server in the client/server exchange coming next.

14.4 Client/Server Exchange

Once the client user has the client/server service ticket, he can establish the session with the server service. The server can decrypt the information coming indirectly from the TGS using its own long-term key with the KDC. The service ticket is then used to authenticate the client user and establish a service session between the server and client. After the ticket's lifetime is exceeded, the service ticket must be renewed to use the service.

The client blindly passes the server portion of the service ticket to the server in the client/server request to establish a client/server session. If mutual authentication is enabled, the target server returns a time stamp encrypted using the service ticket session key. If the time stamp decrypts correctly, not only has the client authenticated himself to the server, but the server also has authenticated itself to the client.

14.5 Remote access - Referral Tickets

A TGT and a service ticket are required to successfully log on to a local system but are also needed to access services on remote computers. The AS and TGS functions are separate within the KDC. This permits the user to use the TGT obtained from an AS in his domain to obtain service tickets from a TGS in other domains.

This is accomplished through referral tickets. Once a trust has been established between two domains,

referral tickets can be granted to clients requesting authorization for services in other domains. When there is a trust established between the two domains, an inter-domain key based on the trust password becomes available for authenticating KDC functions. This can best be explained by example of a user/client seeking services in another domain.

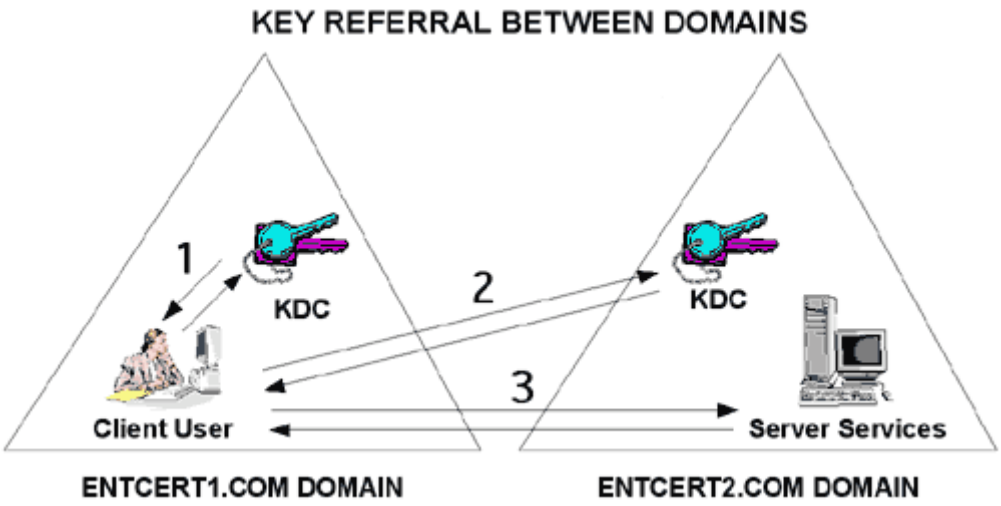


Figure 14.2

As illustrated in Figure 14.2, a user client in Entcert1.com requests authority for a server in Entcert2.com. He utilizes referral tickets. The numbers in Figure 2 correspond to the following numbered explanations:

1. The client contacts its domain KDC TGS using a TGT. The KDC recognizes a request for a session with a foreign domain server and responds by returning a referral ticket for the KDC in the foreign domain.
2. The client contacts the KDC of the foreign domain with the referral ticket. This ticket is encrypted with the inter-domain key. Given that the decryption works, the TGS service for the foreign domain returns a service ticket for the server service in Entcert2.com.
3. The client performs the client/server exchange with the server and begins the user session with the service.

chapter 15 IPsec and Ipv6 security features

The **S**.

15.1 paragraph 1

One

15.2 par 2

One iteration

chapter 16 secure communication – vpn, tls, ssh

Secure communication is when two entities are communicating and do not want a third party to listen in. For that they need to communicate in a way not susceptible to eavesdropping or interception. Internet communications that are based on the Transfer Control Protocol/Internet Protocol (TCP/IP), such as the Hypertext Transfer Protocol (HTTP), Telnet, and File Transfer Protocol (FTP), are not secure because all communication occurs in plaintext. Confidential or sensitive information that is transmitted with these protocols can easily be intercepted and read unless the information is protected by encryption technology.

Considering the recent events, related to NSA surveillance and the deliberate weakening of cryptographic systems, secure communication becomes that much more important.

The primary way the NSA eavesdrops on internet communications is in the network. They have invested in enormous programs to automatically collect and analyze network traffic. They gain access to data through different “secret partnerships” with UK/USA telecommunication providers and not only (tapping undersea cables, intercepting satellite communications and so on). The agency can quickly filter the data, looking for “interesting” traffic ("Interesting" can be defined in many ways: by the source, the destination, the content, the individuals involved, and so on. This data is funneled into the vast NSA system for future analysis).

Secure Web communication protocols provide a way to authenticate clients and servers on the Web and to protect the confidentiality of communication between clients and servers. A variety of secure communication standards that use public key technology have been developed, including Secure Hypertext Transfer Protocol (SHTTP), IP Security (IPSec), PPTP, and L2TP. The leading general-purpose, secure Web communication protocols are SSL 3.0 and the open TLS protocol that is based on SSL. The SSL and TLS protocols are widely used to provide secure channels for confidential TCP/IP communication on the Web.

16.1 VPN

A virtual private network (VPN) extends a private network across a public network, such as the Internet. It enables a computer or network-enabled device to send and receive data across shared or public networks as if it were directly connected to the private network, while benefiting from the functionality, security and management policies of the private network. A VPN is created by establishing a virtual point-to-point connection through the use of dedicated connections, virtual tunneling protocols, or traffic encryption. Major implementations of VPNs include OpenVPN and Ipsec.

Businesses use VPNs to connect remote datacenters, and individuals can use VPNs to get access to network resources when they're not physically on the same LAN (local area network), or as a method for securing and encrypting their communications when they're using an untrusted public network.

A VPN alone is just a way to increase your security and access resources on a network you're not physically connected to. What you choose to do with a VPN is a different story. The most common uses are:

- **Students/workers** need to access resources on their network while they’re at home or travelling.
- **To download files securely**. VPNs are the only way to stay safe when using something like BitTorrent or other Peer To Peer sharing service.
- **To surpass geographically restricted content or services** such as Netflix (video streaming service) or Pandora/Spotify (music streaming services)

Most VPNs rely on tunneling to create a private network that reaches across the Internet. Tunneling is the process of placing an entire packet within another packet before it's transported over the Internet. That outer packet protects the contents from public view and ensures that the packet moves within a virtual tunnel.

This layering of packets is called encapsulation. Computers or other network devices at both ends of the tunnel, called tunnel interfaces, can encapsulate outgoing packets and reopen incoming packets. Users at both ends have to configure the tunnel interfaces they’re responsible for to use a tunneling protocol (also known as encapsulation protocol). A tunneling protocol is a standardized way to encapsulate packets.

The purpose of the tunneling protocol is to add a layer of security that protects each packet on its journey over the Internet. We have to note that the packet is traveling with the same transport protocol it would have

used without the tunnel. A more colorful way to look at the relationship between the protocols would be to think of tunneling as having a package sent to you by a shipping company. The vendor who is sending you the computer packs the computer (**passenger protocol**) in a box (**tunneling protocol**). Shippers then place that box on a shipping truck (**transport protocol**) at the vendor's warehouse (**one tunnel interface**). The truck (**transport protocol**) travels over the highways (**Internet**) to your home (**the other tunnel interface**) and delivers the computer. You open the box (**tunneling protocol**) and remove the computer (**passenger protocol**).

There is no standard that all VPNs follow in terms of their setup but some of the most common equipment used are:

- **Network access server** - a NAS is responsible for setting up and maintaining each tunnel in a remote-access VPN.
- **Firewall** - A firewall provides a strong barrier between your private network and the Internet. IT staff can set firewalls to restrict what type of traffic can pass through from the Internet onto a LAN, and on what TCP and UDP ports.
- **AAA Server** - The acronym stands for the server's three responsibilities: authentication, authorization and accounting. For each VPN connection, the AAA server confirms who you are (authentication), identifies what you're allowed to access over the connection (authorization) and tracks what you do while you're logged in (accounting). An example of AAA server is **RADIUS** (Remote Authentication Dial-in User Service). This protocol isn't just for dial-up users. When a RADIUS server is part of a VPN, it handles authentication for all connections coming through the VPN's NAS.

On a side note, over the passing of time and because of its popularity, companies have developed dedicated VPN devices that business can purchase.

A few examples**: VPN Concentrator** (replace AAA server installed on a generic server), **VPN-enabled/VPN-optimized Router**, **VPN-enabled Firewall** and **VPN Client**.

Encryption is the process of encoding data so that only a computer with the right decoder will be able to read and use it. An encryption key tells the computer what computations to perform on data in order to encrypt or decrypt it. The most common forms of encryption are symmetric-key encryption or public-key encryption: **Symmetric-key encryption** (all users have the same key that is used for both encryption and decryption) and **public-key encryption** (each computer has a public/private key. A computer uses the private key to encrypt while another uses the corresponding public key to decrypt the data).

In a VPN environment, both end of the tunnel encrypt data entering and decrypt it at the other end. But a VPN needs more than just keys to apply the encryption mechanisms. (Protocols come in) A site-to-site VPN could use either Internet protocol security protocol (IPSec) or generic routing encapsulation (GRE). GRE provides the framework for how to package the passenger protocol for transport over the Internet protocol (IP). This framework includes information on what type of packet you're encapsulating and the connection between sender and receiver.

IPSec is a widely used protocol for securing traffic on IP networks, including the Internet. IPSec can encrypt data between various devices, including router to router, firewall to router, desktop to router, and desktop to server. IPSec consists of two sub-protocols which provide the instructions a VPN needs to secure its packets:

- **Encapsulated Security Payload** (**ESP**) encrypts the packet's payload (the data it's transporting) with a symmetric key.
- **Authentication Header** (**AH**) uses a hashing operation on the packet header to help hide certain packet information (like the sender's identity) until it gets to its destination

In a **remote- access VPN**, tunneling typically relies on **Point-to-point Protocol** (PPP) which is part of the native protocols used by the Internet. More accurately, though, remote-access VPNs use one of three protocols based on PPP: **L2F** (Layer 2 Forwarding), **PPTP** (Point-to-point Tunneling Protocol) and **L2TP** (Layer 2 Tunneling Protocol)

16.2 TLS

One problem when you administer a network is securing data that is being sent between applications across an untrusted network. You can use TLS/SSL to authenticate servers and clients and then use it to encrypt messages between the authenticated parties.

History

Netscape Communications created the original specification of secure socket layer (SSL) in 1994, when it became apparent that there was no way to securely transfer reliable protocols across the internet, without fear of interference or snooping of traffic. The first specification, version 1.0 was so heavily criticised by the cryptographic community for the implementation of weak cryptographic algorithms that it was never realised for public use.

The Netscape Communications department revised the specification and released a much improved version 2.0 in February 1995, as described by (Shostack, 1995) the second version of the protocol requested the use of the MD5 hash function and required the use of MD5 for all cipher types, the MD5 algorithm is defined in (RFC 1321).

While SSL version 2.0 was considered a fairly strong and robust protocol, it did have some areas where it was vulnerable. So in 1996 the next iteration of the protocol version 3.0, which was designed by both Netscape and Paul Kocher, was released. As described by (Gibson, 2009) version 3 addressed the implementation of the weak MD5 hash that was implemented in version 2 by producing both an MD5 hash and a SHA-1 hash and XOR'ing the result together to create a hybrid hash that was dependant on both algorithms.

As the protocol was now gaining such traction on the internet, the Internet Engineering Task Force (IETF) took responsibility for the protocol and renamed it to transport Layer Security (TLS) to avoid bias towards any particular company.

Finally the latest and most current version of the TLS standard, 1.2 and was released in August 2008 and has a number of improvements as documented in (RFC 5246), including the removal of older cipher suites like DES and IDEA and the inclusion of the SHA256 cipher suites.

In the authentication process, a TLS/SSL client sends a message to a TLS/SSL server, and the server responds with the information that the server needs to authenticate itself. The client and server perform an additional exchange of session keys, and the authentication dialog ends. When authentication is

completed, SSL-secured communication can begin between the server and the client using the symmetric encryption keys that are established during the authentication process.

For servers to authenticate to clients, TLS/SSL does not require server keys to be stored on domain controllers or in a database, clients confirm the validity of a server's credentials with a trusted root certification authority's (CA's) certificates.

TLS and SSL are most widely recognized as the protocols that provide secure HTTP (HTTPS) for Internet transactions between Web browsers and Web servers. TLS/SSL can also be used for other application level protocols, such as File Transfer Protocol (FTP), Lightweight Directory Access Protocol (LDAP), and Simple Mail Transfer Protocol (SMTP). TLS/SSL enables server authentication, client authentication, data encryption, and data integrity over networks such as the World Wide Web.

The four protocol layers of the SSL protocol (Record Layer, ChangeCipherSpec Protocol, Alert Protocol, and Handshake Protocol) encapsulate all communication between the client machine and the server.

Record Layer

The record layer formats the Alert, ChangeCipherSpec, Handshake and application protocol messages. This formatting provides a header for each message and a hash, generated from a Message Authentication Code (MAC) at the end. The fields that comprise the five-byte header of the Record Layer are: Protocol Definition (1 byte), Protocol Version (2 bytes) and the Length (2 bytes). The protocol messages that follow the header cannot be longer than 16,384 bytes, as specified by the SSL protocol.

ChangeCipherSpec Protocol

The ChangeCipherSpec layer is composed of one message that signals the beginning of secure

communications between the client and server. Though the ChangeCipherSpec Protocol uses the Record Layer format, the actual ChangeCipherSpec message is only one byte long, and signals the change in communications protocol by having a value of '1'.

Alert Protocol

This protocol sends errors, problems or warnings about the connection between the two parties. This layer is formed with two fields: the Severity Level and Alert Description.

Severity Level

The Severity Level sends messages with a '1' or '2' value, depending on the level of concern. A message with a value of '1' is a cautionary or warning message, suggesting that the parties discontinue their session and reconnect using a new handshake. A message with a value of '2' is a fatal alert message, and requires that the parties discontinue their session.

Alert Description

The Alert Description field indicates the specific error that caused the Alert Message to be sent from a party. This field is one byte, mapped to one of twelve specific numbers, and can take on one of the following meanings. Those descriptions that always follow a "fatal" alert message are underlined.

16.3 The handshake protocol in TLS

Messages passed back and forth between the user's browser (client) and web application (server) establish a handshake that begins a secure connection. The following steps are how a SSL handshake is performed. The messages that compose this handshake are: ClientHello, ServerHello, ServerKeyExchange, ServerHelloDone, ClientKeyExchange, ChangeCipherSpec, Finished, ChangeCipherSpec, Finished.

ClientHello

The first message is the ClientHello. Since the client machine is requesting the secure communication session, this message involves a set of options that the client is willing to use in order to communicate with the server. The option categories are: Version of SSL to be used, CipherSuites supported by the client, and CompressionMethods used by the client. Other information that is included in this message is a 32-byte RandomNumber that assists the client in establishing encrypted communications, and a SessionID field that is blank. This message is generated by the client in the web e-mail example when our user wants to check her email and clicks on the "secure connection" option that is made available on many websites.

ServerHello

The second message of the SSL handshake is the ServerHello. In this message, the server makes choices based on the ClientHello message. The server returns five fields, just like the ClientHello message, but fills in the SessionID, and makes firm decisions on the Version of SSL to be used, the CompressionMethod and CipherSuite. The date and time stamp replaces four bytes of the RandomNumber field to avoid repeated random values, and Thomas adds that "the remaining bytes should be created by a cryptographically secure random number generator."

ServerKeyExchange

Now that the server has made decisions for the transmission of data, information must be passed between the parties to determine how data will be encrypted. Since no algorithm has been previously agreed upon, this information is sent with no encryption. This means that all communication for this segment must already be in the public domain. The server's public key is used to encrypt a separate session key to be maintained for this secure communication. Both the client and server will use this same key to encrypt data to be transmitted. To ensure that the communicating parties are who they claim to be, digital certificates are used to provide electronic identification. Digital certificates combine the public key and connect it to the name of the certificate owner. Additionally, these certificates contain public keys to certification authorities like RSA Security or VeriSign and an expiration date so that the person receiving the digital certificate can verify the link between the certificate owner and the certification authority. The certificate only contains the public key, and should never include the private key, else the private key would be compromised, and the entire purpose of having the digital certificate would be voided.

ServerHelloDone

Once the Server has completed the ServerKeyExchange message, the client receives a ServerHelloDone message to indicate that the server is through with its messages. It is similar to a two-way radio conversation when the sending party says “OVER” to announce that he is done sending a message, and signals the receiving party to acknowledge the message that was sent.

ClientKeyExchange

Since SSL does not require a client to have public and private keys in order to establish a SSL session, the ClientKeyExchange message contains information about the key that the client and server will use to communicate. This is the point where the “man in the middle” attack is mitigated since a masquerader must know the server’s private key in order to decrypt this message. This message completes the negotiation processes between the client and the server.

ChangeCipherSpec

The two ChangeCipherSpec messages signal the change of data transmission from an insecure state to a secure state. As each computer sends the ChangeCipherSpec message, it changes its side of the connection into the agreed-upon secure state.

Finished

The two messages signaling the final messages of the SSL handshake ensure that three things are verified before the initial handshake is complete. These are:

- Key Information
- Contents of all previous SSL handshake messages exchanged by the systems
- A special value indicating whether the sender is a client or server

At the end of this handshake process, the user will see a lock icon in the corner of the browser to indicate that a secure protocol has been agreed upon, and is in use by the browser and the web e-mail server.

Message Authentication

Once this information is checked, the communication can continue, appending a message authentication algorithm to the end of each message. Message Authentication is performed by using “an algorithm that uses cryptographic technology to create a digital summary of information so that if the information is altered, the summary (known as a hash) will also change.” (Thomas, 186) MD5 and SHA are common hash functions used in SSL communications.

Resuming a Disconnected Session

If an Alert message disconnects a sessions before the parties are through communicating, that session can be resumed if the client sends a HelloRequest to the server with the properly encrypted SessionID information. The server then determines if the SessionID is valid, exchanges ChangeCipherSpec and Finished messages with the client machine, and secure communication can resume.

TLS/SSL provides numerous benefits to clients and servers over other methods of authentication, including:

- **Strong authentication, message privacy, and integrity -** TLS/SSL can help to secure transmitted data using encryption. TLS/SSL also authenticates servers and, optionally, authenticates clients to prove the identities of parties engaged in secure communication. It also provides data integrity through an integrity check value. In addition to protecting against data disclosure, the TLS/SSL security protocol can be used to help protect against masquerade attacks, man-in-the-middle or bucket brigade attacks, rollback attacks, and replay attacks.
- **Interoperability** - TLS/SSL works with most Web browsers and on most operating systems and Web servers, including the Microsoft Windows operating system, UNIX, Novell, Apache, Netscape Enterprise Server, and Sun Solaris. It is often integrated in news readers, LDAP servers, and a variety of other applications.
- **Algorithm flexibility** - TLS/SSL provides options for the authentication mechanisms, encryption algorithms, and hashing algorithms that are used during the secure session.

- **Ease of use** - Because you implement TLS/SSL beneath the application layer, most of its operations are completely invisible to the client. This allows the client to have little or no knowledge of the security of communications and still be protected from attackers.

There are a few limitations to using TLS/SSL, including: **Increased processor load** (This is the most significant limitation to implementing TLS/SSL. Cryptography, specifically public key operations, is CPU-intensive. As a result, performance varies when you are using SSL. Unfortunately, there is no way to know how much performance you will lose. The performance varies, depending on how often connections are established and how long they last. TLS uses the greatest resources while it is setting up connections) and **Administrative overhead** (A TLS/SSL environment is complex and requires maintenance; the system administrator must configure the system and manage certificates)

16.4 SSH

One Secure Shell, or SSH, is a cryptographic (encrypted) network protocol for initiating text-based shell sessions on remote machines in a secure way.

This allows a user to run commands on a machine's command prompt without them being physically present near the machine. It also allows a user to establish a secure channel over an insecure network in a client-server architecture, connecting an SSH client application with an SSH server.

History

SSH1 and the SSH-1 protocol were developed in 1995 by Tatu Ylönen, a researcher at the Helsinki University of Technology in Finland. After his university network was the victim of a password-sniffing attack earlier that year, Ylönen whipped up SSH1 for himself. When beta versions started gaining attention, however, he realized that his security product could be put to wider use.

In July 1995, SSH1 was released to the public as free software with source code, permitting people to copy and use the program without cost. By the end of the year, an estimated 20,000 users in 50 countries had adopted SSH1.

Also in 1995, Ylönen documented the SSH-1 protocol as an Internet Engineering Task Force (IETF) Internet Draft, which essentially described the operation of the SSH1 software after the fact. It was a somewhat ad hoc protocol with a number of problems and limitations discovered as the software grew in popularity. These problems couldn't be fixed without losing backward compatibility, so in 1996, SCS introduced a new, major version of the protocol, SSH 2.0 or SSH-2, that incorporates new algorithms and is incompatible with SSH-1. In response, the IETF formed a working group called SECSH (Secure Shell) to standardize the protocol and guide its development in the public interest. The SECSH working group submitted the first Internet Draft for the SSH-2.0 protocol in February 1997.

In 1998, SCS released the software product "SSH Secure Shell" (SSH2), based on the superior SSH-2 protocol. However, SSH2 didn't replace SSH1 in the field, for two reasons. First, SSH2 was missing a number of useful, practical features and configuration options of SSH1. Second, SSH2 had a more restrictive license.

This situation promises to change, however, as a result of two developments: a loosening of the SSH2 license and the appearance of free SSH-2 implementations.

On Unix-like systems, the list of authorized public keys is typically stored in the home directory of the user that is allowed to log in remotely, in the file ~/.ssh/authorized_keys. This file is respected by ssh only if it is not writable by anything apart from the owner and root. The ssh-keygen utility produces the public and private keys, always in pairs. SSH also supports password-based authentication that is encrypted by automatically generated keys. In this case the attacker could imitate the legitimate server side, ask for the password, and obtain it (man-in-the-middle attack). However, this is possible only if the two sides have never authenticated before, as SSH remembers the key that the server side previously used. The SSH client raises a warning before accepting the key of a new, previously unknown server. Password authentication can be disabled.

SSH provides multiple mechanisms for authenticating the server and the client. Two of the commonly used authentication mechanisms are password based, and key based authentication. Although password based authentication is also secure, it’s advisable to use key based authentication instead.

SSH protocol version 2 is the default protocol used these days. This is due to some major advancements in version 2 compared to version 1. The workflow of the SSH login is almost same as that of version 1,

however there are some major changes done in the protocol level. Some of these changes include improved encryption standards, Public key certification, much better message authentication codes, reassignment of session key(which is altered every hour) etc.

Some of the most important characteristics of SSH are:

- **Privacy communication** - means that the connection, which provides a remote shell login, must be encrypted to prevent eavesdropping.
- **Integrity check** – There must be a mechanism that checks whether the data sent by either sides was not altered or tampered with.
- **Identity of both server and client must be provided to each other**.
- SSH Tunneling
- TCP port forwarding

How does SSH work

The SSH protocol employs a client-server model to authenticate two parties and encrypt the data between them.

The server component listens on a designated port for connections. It is responsible for negotiating the secure connection, authenticating the connecting party, and spawning the correct environment if the credentials are accepted.

The client is responsible for beginning the initial TCP handshake with the server, negotiating the secure connection, verifying that the server's identity matches previously recorded information, and providing credentials to authenticate.

An SSH session is established in two separate stages. The first is to agree upon and establish encryption to protect future communication. The second stage is to authenticate the user and discover whether access to the server should be granted.

Negotiating Encryption for the Session

When a TCP connection is made by a client, the server responds with the protocol versions it supports. If the client can match one of the acceptable protocol versions, the connection continues. The server also provides its public host key, which the client can use to check whether this was the intended host.

At this point, both parties negotiate a session key using a version of something called the Diffie-Hellman algorithm. This algorithm (and its variants) make it possible for each party to combine their own private data with public data from the other system to arrive at an identical secret session key.

The session key will be used to encrypt the entire session. The public and private key pairs used for this part of the procedure are completely separate from the SSH keys used to authenticate a client to the server.

The basis of this procedure for classic Diffie-Hellman is:

1. Both parties agree on a large prime number, which will serve as a seed value.
2. Both parties agree on an encryption generator (typically AES), which will be used to manipulate the values in a predefined way.
3. Independently, each party comes up with another prime number which is kept secret from the other party. This number is used as the private key for this interaction (different than the private SSH key used for authentication).
4. The generated private key, the encryption generator, and the shared prime number are used to generate a public key that is derived from the private key, but which can be shared with the other party.
5. Both participants then exchange their generated public keys.
6. The receiving entity uses their own private key, the other party's public key, and the original shared prime number to compute a shared secret key. Although this is independently computed by each

party, using opposite private and public keys, it will result in the same shared secret key.

7. The shared secret is then used to encrypt all communication that follows.

The shared secret encryption that is used for the rest of the connection is called binary packet protocol. The above process allows each party to equally participate in generating the shared secret, which does not allow one end to control the secret. It also accomplishes the task of generating an identical shared secret without ever having to send that information over insecure channels.

The generated secret is a symmetric key, meaning that the same key used to encrypt a message can be used to decrypt it on the other side. The purpose of this is to wrap all further communication in an encrypted tunnel that cannot be deciphered by outsiders.

After the session encryption is established, the user authentication stage begins.

Authenticating the User's Access to the Server

The next stage involves authenticating the user and deciding access. There are a few different methods that can be used for authentication, based on what the server accepts.

The simplest is probably password authentication, in which the server simply prompts the client for the password of the account they are attempting to login with. The password is sent through the negotiated encryption, so it is secure from outside parties.

Even though the password will be encrypted, this method is not generally recommended due to the limitations on the complexity of the password. Automated scripts can break passwords of normal lengths very easily compared to other authentication methods.

The most popular and recommended alternative is the use of SSH key pairs. SSH key pairs are asymmetric keys, meaning that the two associated keys serve different functions.

The public key is used to encrypt data that can only be decrypted with the private key. The public key can be freely shared, because, although it can encrypt for the private key, there is no method of deriving the private key from the public key.

Authentication using SSH key pairs begins after the symmetric encryption has been established as described in the last section. The procedure happens like this:

1. The client begins by sending an ID for the key pair it would like to authenticate with to the server.
2. The server check's the authorized_keys file of the account that the client is attempting to log into for the key ID.
3. If a public key with matching ID is found in the file, the server generates a random number and uses the public key to encrypt the number.
4. The server sends the client this encrypted message.
5. If the client actually has the associated private key, it will be able to decrypt the message using that key, revealing the original number.
6. The client combines the decrypted number with the shared session key that is being used to encrypt the communication, and calculates the MD5 hash of this value.
7. The client then sends this MD5 hash back to the server as an answer to the encrypted number message.
8. The server uses the same shared session key and the original number that it sent to the client to calculate the MD5 value on its own. It compares its own calculation to the one that the client sent back. If these two values match, it proves that the client was in possession of the private key and the client is authenticated.

A short list of differences between SSH v1 and SSH v2:

- Diffie-Hellman key is used instead of the server key for sharing the session key in v2 protocol
- No Rhosts support in SSH v2

- SSH protocol version 1 only allows negotiation of the symmetric encryption algorithm, all other things are hard corded(mac, compression etc)
- SSH 2 supports certificates for public keys used
- A SSH 2 server can dictate the client to use multiple authentication methods in a single session to succeed. However SSH v1 only supports one method per session
- SSH version 2 allows the change of session key periodically.

chapter 17 Secure storage

The ever-increasing amount of valuable digital data both at home and in business needs to be protected, since its irrevocable loss is unacceptable. Cloud storage services promise to be a solution for this problem. In this study we have examined the security mechanisms of seven cloud storage services: CloudMe, CrashPlan, Dropbox, Mozy, TeamDrive, Ubuntu One, and, Wuala.

We identified three categories of security requirements:

Transport Security (to secure communication between client and server)

- communication confidentiality and integrity
- server authentication
- suitable cryptography

Encryption (to disable the provider to examine stored data)

- client-side encryption of data
- client-side encryption of file names
- non-deterministic generation of encryption keys
- suitable cryptography

Secure File Sharing (to protect documents shared by a closed group)

- clear description which flavor of sharing is used
- obfuscated link
- no indexing by external search engines
- reversible sharing
- uninvited users are excluded by cryptographic means

17.1 Transport Security

Cloud storage providers usually provide client software which assists users in setting up their synchronization or backup schemes on the local devices. The actual transmission of all data with the remote storage servers is also handled by the client software. The server must authenticate itself to the client and all communication should be encrypted and its integrity ensured.

It is important to use appropriate cryptographic functions. All primitives, like symmetric and asymmetric encryption functions and hash functions should be up to date. This includes the algorithms as well as their parameters, like key lengths. If keys are generated this should be done by a secure high-entropic key generator.

Algorithms and protocols should always be public, as stated by Kerckho's principle". Keeping these things secret is always a risk, that does not increase security but decreases it dramatically. Developing a cryptographic protocol is a very difficult task. In the past, even protocols designed by well-respected experts have failed. So it is in most cases a bad idea to invent a new algorithm for a well known problem, especially if a widely accepted solution is available.

The standard protocol TLS offers an established solution for transport security. There should be severe reasons for replacing it by something else for the same ask.

17.2 Encryption

The main reason to use a cloud storage provider, for both individuals and companies, is to always have a

backup of valuable data which is off-premises yet easily accessible. The data itself should be protected in such a way that even in the event of a successful attack, the contents of the stored data remain confidential. To this end, all data needs to be stored on the remote servers in encrypted form.

There are several cryptographically secure encryption schemes available which can be used freely. Cloud storage providers often offer a general encryption of all data stored on their servers using a company key which is known only to them. This may prevent data theft from external attackers, but does not protect against any attacks which include theft of the encryption key or internal attacks conducted by personnel who are able to gain access to these keys. Therefore, all data should be encrypted on the client system before the data is transmitted into the cloud using a key unknown to the service provider.

Standalone software may be used to encrypt all data on the client system, but this has drawbacks: The software has to be installed, administrated and operated on all client systems in addition to the client software of the cloud storage provider. The key used to encrypt the data needs to be distributed to all devices which are used to access the stored data. In the event that this key is lost, the data can never be decrypted again. As a precautionary measure, all keys used to encrypt data could be integrated into some kind of key escrow system to guard against data loss.

All keys that are used for encryption should be generated at (pseudo) random. This requirement ensures that two cryptograms of the same clear text are different.

17.3 Secure File Sharing

Sharing files appears in three different flavors: Sharing files with other subscribers of the same service, Sharing files with a closed group of non-subscribers, Sharing files with everybody.

In any case the service should describe clearly which flavor of sharing is used.

The files that are being shared should only be accessible to the closed user group that was decided by the sharing user. It should also be possible to revert sharing for each individual file. A list of files currently being shared by the user could be accessed in the web interface or in the client application. It should be possible to deal with different access rights and at any time the sharing user should be able to grant, edit or remove individual access rights. If client-side encryption is used, sharing files should not weaken the security level. In particular, the cloud storage service provider should not be able to read shared files. If client-side encryption is used, a uninvited user should be excluded by cryptographic means from the closed user group. In particular, this means that an encryption key that is known by the uninvited user can no longer be used for the encryption of new files.

Sharing files with everybody has the security requirement to hide informations about user names. If there is no client-side encryption with keys individual to the user the service knows which clients share files even though the clients do not use file sharing feature.

17.4 CloudMe

Transport Security

- CloudMe does not encrypt the data transferred between the server and the client.

Encryption

- CloudMe does not encrypt the files that are stored on the server. Since the communication between the client and the server is also not encrypted, attackers are able to intercept every file a user uploads to the service.

Sharing

- The required password length of one character is not enough to guard against any attacks.

17.5 CrashPlan

Transport Security

- CrashPlan does not use SSL/TLS to secure the communication between a client and the CrashPlan server, instead a self-made, unpublished protocol is used.
- The communication between the client and other backup destinations is not secured by SSL/TLS. This is a disadvantage if these destinations are outside of an intranet.

Encryption

- The key used to encrypt the files is chosen at random during the installation of the software and will be referred to as data key.
- CrashPlan provides multiple options to encrypt files, which are explained in high detail on the CrashPlan website: Securing the data encryption key with the account password, Securing data encryption key with a private password or Using an exclusively local stored data encryption key.
- With the default option, it is possible for CrashPlan to decrypt and access the data stored on their servers, since both the data key and the password used to secure the data key are known to CrashPlan.
- With the second option, CrashPlan can't access the encrypted data unless the user is using the web restore function, where he has to enter his private password which is then used to unlock the data key.
- Using the third option, the private data key has to be entered when using the web restore function. The user is responsible to store this key in a secure and safe way.

Sharing

- CrashPlan does not support file sharing or file publication. When using the friend feature to store files on the computers of other users, these files are stored encrypted and are not accessible to the other users.

17.6 Dropbox

Transport Security

- Dropbox uses TLS to encrypt the communication between the client application and the server. The communication between the browser and the web interface is encrypted by using HTTPS.

Encryption

- Dropbox uses AES-256 to encrypt data stored on its servers. The data will not be encrypted at the client; instead Dropbox encrypts the data after the upload on the server-side using its own encryption key.
- While the encryption of data in transit meets the requirements, Dropbox has not optimally implemented the encryption of the stored data. Since Dropbox itself encrypts the data on the server-side, users cannot be sure by cryptographic means that all stored data is highly confidential.

Sharing

- Sharing files with subscribers meets our previously described requirements. Dropbox has some problems when sharing files with non-subscribers / everybody.
- Using a simple script which iterated through possible URL combinations we were able to search for the existence of specific files inside the Public folder.

17.7 Mozy

Transport Security

- Mozy uses TLS to encrypt the communication between the client application and the server. The communication between the browser and the web interface is encrypted by using HTTPS.

Encryption

- All data is generally encrypted at the client, before being transferred to the Mozy server. The user can select between two encryption methods. The default is to use an 448-bit Blowfish encryption key which is provided by and therefore known to Mozy. As an alternative, the user can use a personal 256-bit AES encryption key.

Sharing

- Mozy does not offer to share files with other people.

17.8 TeamDrive

Transport Security

- The web interface uses HTTPS to secure the communication between browser and server. The communication between clients and the TeamDrive server uses HTTP, enhanced by a self-made, unpublished protocol.

Encryption

- TeamDrive uses AES-256 for file encryption. The data is encrypted at the client before it is transmitted to the server. Every space uses an individual AES key for file encryption. These AES keys are not based on a password and are not known to TeamDrive, therefore TeamDrive is not able to access any data stored by its users on their servers.

Sharing

- Sharing files is supported by cryptographic means. When sharing files with another subscriber, the TeamDrive server sends the public key of the invitee. The inviting user encrypts the AES key of the space with this key. In doing so, he trusts that the received key is authentic. An invitation including the encrypted space key is sent to the invitee. After decrypting the space key with his secret key the invitee can access and decrypt all files inside the space.

17.9 Ubuntu One

Transport Security

- Ubuntu One uses SSL/TLS to encrypt the communication between the clients and the server. The communication between the browser and the web interface is encrypted by using HTTPS.

Encryption

- Ubuntu One does neither encrypt data using the client software nor on the server. Thus, the data itself is not protected against unauthorized access from attackers who successfully circumvent authentication security of the service. Ubuntu makes the missing encryption very clear in their FAQ.

Sharing

- Sharing files with subscribers. The sharing of files between registered Ubuntu One users meets our requirements. Sharing files with everybody: The URL of a published file consists of a mix of numbers and upper- and lower-case characters. The URL does not contain a user name which impedes information gathering.

17.10 Wuala

Transport Security

- Wuala uses a proprietary client / server-communication protocol instead of the standardized and well-known SSL/TLS protocol to secure the communication between a client and the Wuala server. According to Wuala, integrity checks are used to protect transmitted data in transit. In combination with the convergent encryption scheme employed by Wuala, the absence of encryption during transmit allows attackers to sniff exchanged messages and attempt information gathering attacks.

Encryption

- The idea behind Wuala's encryption scheme is an untrusted file system that is secured by cryptographic methods. The employed system is an implementation of a folder tree structure for cryptographic file systems called Cryptree that has been published by Grolimund from ETH Zurich. The trust anchor is a symmetric root key r which is derived from the user's password. Wuala calculates individual keys for every directory and individual keys for every file. All of them are accessible via r. They can be given to partners in order to share data.

Sharing

- Security of shared files depends on the invitee. Sharing between registered users meets all mandatory requirements. The files are not readable by Wuala. When sharing files with another subscriber, the Wuala server sends the public key of the invitee to the inviting user. He encrypts a key for the invitee. The result is sent via Wuala to the invitee. In doing so, the inviting user trusts that the received keys are authentic. Sharing files with non sub-subscribers is based on secret web links. Knowing the link is equal to having the right to access the file. The value included in the URL is sufficiently large, appears to be random and is only valid for this folder. The files shared with this method are not indexed by search engines, and sharing can be reversed anytime.

17.11 Conclusion

The study shows that most of the analyzed cloud storage providers are aware of the extreme importance of data security and privacy. Nevertheless, none of the examined cloud storage providers meets all mandatory security requirements.

Transport Security was a problem for CrashPlan, TeamDrive and Wuala because they deny the usage of SSL/TLS. Instead they use unpublished, self-made protocols { a very error-prone approach. CloudMe does not take any measure to protect the security of files during transmission.

Encryption was a problem for CloudMe, Dropbox and Ubuntu One because they do not use client-side encryption, thus the provider is able to read the data. Mozy does not encrypt filenames. The convergent encryption scheme used by Wuala enables attacks by a server-side attacker.

Sharing of data was a problem for CloudMe, Dropbox, TeamDrive and Wuala. Problems occur if files are shared with non-subscribers on the principle of a long, unpredictable URL. CloudMe does not obfuscate this URL adequately. Dropbox gives an unclear description with regard to sharing details, TeamDrive is weak when uninviting a group member and Wuala enables information gathering by including the user name in public URLs. CloudMe does not prevent search engines from accessing the workspace.

This study, as presented by the Fraunhofer Institute for Secure Information Technology, is not meant to nominate the best cloud storage service that fits all needs of any possible user. This is impossible. Instead, we want to give some advice, that may help selecting a service for a particular use case. First of all, evaluate your use case, make clear, which problem you want to solve by using a storage service. Align your requirements to the features of the examined services. In addition to concrete security requirements it is recommendable to observe some extra aspects. It is worthwhile to consider using more than one service to reduce the impacts of service downtime. Further, calculation of the time to recover all data from the cloud is recommended. Depending on the individual amount of data, this may take several days. Having a plan for a provider change in the future reduces the dependency on a particular provider (provider lock-in). This will be relevant, for example, if the chosen provider is getting to expensive or is not longer compliant with governmental rules.

chapter 18 P2P security

18.1 introduction

Peer-to-Peer (P2P) networking is a fairly popular concept. Networks such as BitTorrent and eMule make it easy for people to find what they want and share what they have. For one thing, sharing files on your computer with anonymous and unknown users on the general public Internet goes against many of the basic principles of securing your computer. For one thing, sharing files on your computer with anonymous and unknown users on the general public Internet goes against many of the basic principles of securing your computer.

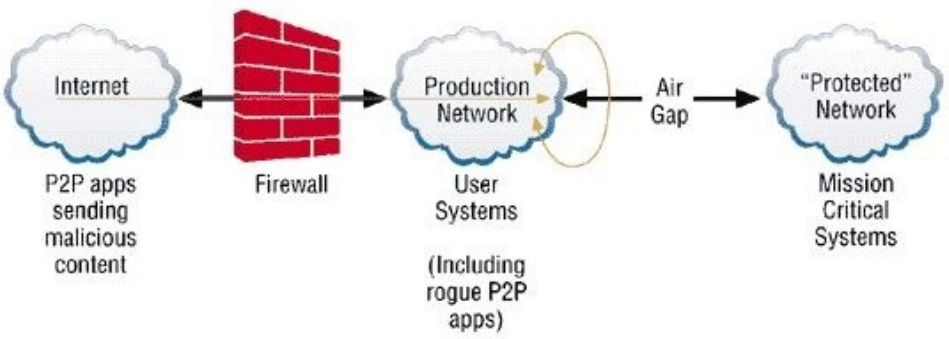
However, in order to share files on your computer and sometimes in order for you to access files on other computers within a P2P network such as BitTorrent, you must open a specific TCP port through the firewall for the P2P software to communicate. In effect, once you open the port you are no longer protected from malicious traffic coming through it.

More technically, a P2P network is a special type of computer network that exhibits self-organization, symmetric communication, and distributed control. The network is self-organizing in that there is typically no centralization of resources. As a result, link capacity is typically distributed throughout peers in the network, and as a result control is distributed, as well.

18.2 need for security

In these turbulent times you would think that P2P security would be the least of the world's problems. However corporate fraud and loss of revenue due to attacks on their internal networks has brought P2P to the forefront in the IT world. Napster was the headliner but since its high profile court case more and more P2P applications have been causing the corporate world headaches, which it could do without. With better security protocols this headache could be turned into a valuable asset for the corporate world and for the world.

The diagram on the next page illustrates the gaps in security when using P2P applications. We can see that we are letting these applications get inside our networks. The security of our "secure" network is now in jeopardy.



Following on from this, is the question of what must we protect ourselves against. We must outline the elements that our important to use, before we address the issue of the security. The main points of this are connection control, access control, operation control, anti-virus, and of course the protection of the data stored on our machines.

18.3 security mechanisms

One All security mechanisms deployed today are based on either symmetric/secret key or asymmetric/public key cryptography, or sometimes a combination of the two. Here we will introduce the basic aspects of the secret key and public key techniques and compare their main characteristics.

Secret Key Techniques:
Secret key techniques are based on the fact that the sender and recipient share a secret, which is used for various cryptographic operations, such as encryption and decryption of messages and the creation and verification of message authentication data. This secret key must be exchanged in a separate out of bound procedure prior to the intended communication (using a PKI for example).

Public Key Techniques:
Public Key Techniques are based on the use of asymmetric key pairs. Usually each user is in possession of just one key pair. One of the pair is made publicly available, while the other is kept private. Because one is available there is no need for an out of band key exchange, however there is a need for an infrastructure to distribute the public key authentically. Because there is no need for pre-shared secrets prior to a communication, public key techniques are ideal for supporting security between previously unknown parties.

Asymmetric Key Pairs:
Unlike a front door key, which allows its holder to lock or unlock the door with equal facility, the public key used in cryptography is asymmetric. This means just the public key can encrypt a message with relative ease but decrypt it, if at all, with considerable difficulty.

Besides being one-way functions, cryptographic public keys are also trapdoor functions- the inverse can be computed easily if the private key is known.

18.4 protocols

Mechanisms for establishing strong, cryptographically verifiable identities are very important. These are industry standard authorization protocols that allow peers to ensure that they are speaking with the intended remote system.

Secure Sockets Layer (SSL) protocol:
For protection of information transmitted over a P2P network, some P2P's employ the industry-standard Secure Sockets Layer (SSL) protocol. This guarantees that files and events sent will arrive unmodified, and unseen, by anyone other than the intended recipient. Moreover, because both peers use SSL both sides automatically prove who they are to each other before any information is transferred over the network. The protocol provides mechanisms to ensure tamperproof, confidential communications with the right counterpart, using the same, well-proven techniques used by all major website operators to protect consumer privacy and financial information transmitted on the Internet.

IPSec technologies:
Most VPNs (virtual private networks) use IPSec technologies, the evolving framework of protocols that has become the standard for most vendors. IPSec is useful because it is compatible with most different VPN hardware and software, and is the most popular for networks with remote access clients. IPSec requires very little knowledge for clients, because the authentication is not user-based, which means a token (such as Secure ID or Crypto Card) is not used. Instead, the security comes from the workstation's IP address or its certificate (e.g. X.509), establishing the user's identity and ensuring the integrity of the network. An IPSec tunnel basically acts as the network layer protecting all the data packets that pass through, regardless of the application.

Public Key Infrastructure (PKI) An industry standard:
A full-featured X.509 Public Key Infrastructure (PKI) over a Secure Sockets Layer (SSL) network backbone - the combination of X.509 PKI authentication and SSL transport encryption is the established cryptographic standard for Internet e-commerce.

Use of X.509 PKI authentication allows security certificates from Endeavors, or from any other recognized X.509 certificate authority, to be used to establish the true identity of any peer device when it comes on-line. Use of SSL point-to-point security encryption enables each pair of peers that communicate with each other to have a unique key for that pairing. The advantage of SSL encryption is that when a peer goes off-line from a community, all its unique pairing keys become invalid, but no pairing keys between other members of the community are affected.

What about VPN Security?
The key word in "virtual private networks" is private. The last thing a business wants is to have sensitive corporate information end up in the hands of some hacker, or worse, the competition. Fortunately, VPNs are widely considered extremely secure, despite using public networks.

Why are they secure?

In order to authenticate the VPNs users, a firewall will be necessary. All VPNs require configuration of an access device, either software- or hardware-based, to set up a secure channel. A random user cannot simply log in to a VPN, as some information is needed to allow a remote user access to the network, or to even begin a VPN handshake. When used in conjunction with strong authentication, VPNs can prevent intruders from successfully authenticating to the network, even if they were able to somehow capture a VPN session.

18.5 the future of P2P security

The constant running theme in the security of P2P is that of trust. Trust in the other users who we interact with, and trust within the software vendors who supply us with the necessary applications. If we could have more faith in this trust, or feel a greater sense of security, maybe the development of P2P would grow even faster than it is already doing.

Many proposals are already being studied. People are acknowledging that security is an area P2P must address, if it is to be accepted by consumers.

Users Gaining Their Own Trust:

One very interesting idea recently proposed, is that of users gaining trust within the P2P community. All users would be assigned a unique digital signature, like IP, but per user and not per machine. Associated with this digital signature would be a level of trust. Trust levels would vary from say zero, to twenty. Depending on a users behaviour in the past, their trust level would either be promoted on the grounds of valid use of the network, of demoted with acts of malice and misuse.

The proposed plan states that all users trust level would begin at a rather low level. This is merely to combat unwanted users creating new accounts, and abusing the new high trust level immediately. Users would have to be active on the network for some time (say one/two months), before their trust level would be pushed up a level. Users could also keep a local record of other known users, to which they may want to share a local trust level, and bypass the global trust policy.

This proposal has many hurdles to jump of course. It is merely an idea to be developed. The problem that it overcomes is that of the centralized managing authority. Instead, the users of the network are the authority. If the general public continuously try to demote a user, he/she will eventually lose all their privileges, and become silenced from other users. This idea also rewards genuine users, for their efforts in keeping the network policed, and for their good behaviour on the network.

The idea is possibly a bit too naive, as we all know that must humans(especially adolescent ones), will do the exact opposite of what they are meant to do, if given no choice. In other words, people do not like to be told what to do.

Biometrics:

Biometrics involves the use of a person’s unique characteristics to authenticate them. Traits that are commonly utilized include a person’s facial image, signature, fingerprint or retinal pattern. One key feature of biometrics is that the user is no longer required to remember any passwords or store any key data, a major weakness in conventional authentication systems.

Ultimately, the technology could find its strongest role as an integrated and complementary piece of a larger authentication system, perhaps in combination with the cryptographic certificates mentioned above, rather than a stand-alone single point of defense.

In the future, many experts foresee biometrics both playing a key role in enabling public key infrastructure deployment by protecting public and private keys and residing in smart card technology in an effort to support personalized e-commerce.

Quantum Key Cryptography:

For the short term, The US Government is adopting a new encryption standard called Advanced Encryption Standard (AES), which will eventually replace DES. "When approved, the AES will be a public algorithm designed to protect sensitive government information well into the 21st century." If that's true, what will be used after AES?

One idea currently being proposed is the notion of Quantum Cryptography. Many modern encryption

systems depend on the difficulty in mounting brute force attacks on secret keys, due to processing and time constraints. Although still at the theoretical stage, the performance improvements given by a hypothetical quantum computer would render many algorithms useless.

Obviously new encryption algorithms would be needed. Quantum encryption uses photon state as the key for encoding information. According to the Heisenberg uncertainty principle, it's impossible to discover both the momentum and position of a particle at any given instant in time. Therefore, in theory, an intruder can't discover secret keys based on particle state information; the intruder would need the actual particle to decipher any data encrypted with a key.
Unfortunately this concept is, for the moment, incredibly complex to implement. IBM scientists constructed the first working prototype of a quantum key distribution (QKD) system in the late 80's. Back then they could transmit quantum signals just under half a meter through open air. Today, fiber optic cables can transmit the signal up to 31 miles. This still isn't very far, but it is definitely good progress. And although we might not see QKD come to market for quite some time, the technology sounds incredibly promising.

18.6 key points to consider when using a P2P network

1. **Don't Use P2P on a Corporate Network** : At least, don't ever install a P2P client or use P2P network file sharing on a corporate network without explicit permission- preferably in writing. Having other P2P users downloading files from your computer can clog the company's network bandwidth. That is the best-case scenario. You may also inadvertently share company files of a sensitive or confidential nature. All of the other concerns listed below are also a factor.
2. **Beware The Client Software:** Installing the software might cause system crashes or problems with your computer in general. Another factor is that the client software is typically hosted from every participating user's machine and could potentially be replaced with a malicious version.
3. **Don't Share Everything :** Many users unknowingly designate the root "C:" drive as their shared files folder which enables everyone on the P2P network to see and access virtually every file and folder on the entire hard drive, including critical operating system files.
4. **Scan Everything :** You should treat all downloaded files with the utmost suspicion. As mentioned earlier, you have virtually no way of ensuring that what you downloaded is what you think it is or that it doesn't also contain some sort of Trojan or virus.

18.7 conclusion

It is obvious from the above that security is a crucial issue when it comes to designing and implementing P2P systems. At the moment it is probably the main inhibiting factor for the growth of P2P. It is vital that users become confident in the ability of the security measures being utilized to protect them, in order for P2P technology to reach its full potential. At the moment, security measures in general are failing to inspire consumer confidence, a problem that must be addressed immediately.

chapter 19 personal profiles, data verification

The **S**.

19.1 paragraph 1

One

19.2 par 2

One iteration

chapter 20 electronic vote

The **S**.

20.1 paragraph 1

One

20.2 par 2

One iteration

chapter 21 electronic payments

The **S**.

21.1 paragraph 1

One

21.2 par 2

One iteration

chapter 22 smart cards

The **S**.

22.1 paragraph 1

One

22.2 par 2

One iteration

chapter 23 biometrics

The **S**.

23.1 paragraph 1

One

23.2 par 2

One iteration

chapter 24 crypto currencies

The **S**.

24.1 paragraph 1

One

24.2 par 2

One iteration