Computer security

Connecting an insecure computer to a network places that computer, its users and any information it contains at risk. Insecure computers also represent a threat to other computers, users and information on the network, since intruders frequently use one compromised machine to attack others either simply by using it to generate network traffic in a distributed denial of service attack, or more subtly by using the access that the compromised computer or its users have to compromise others.

Securing networked computers, as described in this chapter, should be the first goal of any security plan. Network security measures, described in the next chapter, are a very useful additional layer of protection and may sometimes be the only thing that can be done. However if a computer itself is not secure then there will always be ways to get around network security measures and attack it directly.

Different security measures are applicable to server systems and to user client systems. This chapter looks first at measures that should be used on all computers, then separately at some key points for user client computers and for servers. Note, however, that many computers are capable of performing both roles - laptops can run network services while large server boxes can also support interactive users. Security plans should not assume that there is a clear division between the types and be ready to apply server-style measures to small portable devices and client-style measures to large fixed ones if that is how they are actually used.

3.1 Security Measures on All Systems

3.1.1 Patches

All software - including operating systems, applications programs and low-level drivers - contains bugs. A few of those bugs will have implications for the security of the computer. Exploiting bugs in network services is the most common form of technical attack by intruders. The easiest way to reduce the effect of these bugs is not to run unnecessary programs or operating system services - if a service is not running then its bugs do not matter. For those services and operating system components that must be used and exposed to attack, it is essential to ensure that their configurations are secure and that they are updated to deal with any new security problems that may be discovered.

Most equipment, software and operating system vendors provide patches to fix security bugs. Increasingly these are made available on regular schedules and with automated tools available to determine whether a computer needs patching and, if so, to do it. Any computer that may be visible from Janet or the Internet should be set to receive these automated patches, either direct from the vendor or from a local patch management service. If there are particular reasons for a particular computer not to install patches automatically, for example because of potential incompatibilities with other software, then additional security measures will be needed to protect it until the patch can be installed manually. Attacks on vulnerable computers generally increase once a patch is released, both because the patch gives intruders more information and because those who already knew about the vulnerability have less reason to hide their knowledge.

Documents are now available for most operating systems suggesting good practice for running them in a networked environment. A selection of these is included in the references in section 7 of this guide. Mailing lists can also be useful sources of up-to-date information. It is particularly recommended that those responsible for systems subscribe to the alert lists maintained by their operating system and software vendors. Some of the checklists for maintaining secure systems have been implemented as scripts or programs. These can be used to improve the default security level of a system - removing obviously insecure permissions and poor configurations, for example - but they do not remove the need to check and update computers as new problems are found. If a script of this kind is available from a trusted source for your system, then it may be worth using it as a starting point.

3.1.2 Privileged Accounts

A common problem, which greatly increases the severity of a security breach, is when a privileged account (administrator, root, etc.) is compromised. This is likely to give the malicious intruder or their malware complete control of the computer or even, if the account's privileges include modification of central directories or configurations, every computer on the network. Such problems can result both from technical attacks (e.g. finding a weak privileged account password or a vulnerable program running under that account) and human ones (e.g. persuading an administrator or their computer to run malware). Privileged accounts also increase the impact of mistakes: if an unprivileged user attempts to delete files in the wrong directory then they may be protected by file and directory privileges, a privileged user's mistake can easily delete an entire filesystem. Privileged accounts therefore need to be configured and used very carefully.

The most common carrier for malicious code is the world wide web and, to some degree, e-mail. Clicking on the wrong attachment or just browsing to the wrong webpage (if scripts are enabled in the web browser) can result in an instant compromise. Reading e-mail or browsing the web as an administrator account is therefore doubly dangerous. Some policies suggest that e-mail and web clients should simply be removed from privileged accounts, though this may make processes such as downloading and installing software more cumbersome. It seems at least advisable to restrict functions such as automatic execution of scripts in client programs and other applications (such as office programs) that they may initiate. Browsing with administrative privileges should be limited to sites known to be trusted, such as those providing software downloads.

Most operating systems now have the ability to run a single command or function with administrator privileges, rather than needing to log in to a full administrator account. Such facilities may well be a safer alternative for operations that can be identified and configured in advance. Where full administrator access is required, this should only be used for the tasks that actually need administrator access, and not for general work activities. Since the consequences of actions taken as administrator can be so dramatic, detailed logging of what is done by those accounts may help to understand and mitigate the consequences of a mistake or malicious use. In particular it should always be possible to determine which individual is using administrator privileges as their memory of what happened is likely to be vital evidence both in recovering from an incident and preventing it recurring. Where possible, each administrator account should be assigned to a single individual; if the operating system does not permit this then administrators should have to log in using their own account first, before adopting the administrator role, to provide this traceability.

One of the easiest ways to break into an administrator account is through a weak password, or one that has not been changed from its default. Any new equipment or software should be checked for default passwords, which should always be changed or the account removed. Passwords for administrator accounts should always follow best practice for length, complexity and management; administrator access to key information systems may well justify the use of two-factor authentication, as discussed in the section on Remote Login below.

3.1.3 Preparing for Incidents

No security measure is infallible, so organisations should always prepare for when (not if) a security incident occurs. Being able to detect incidents promptly and respond to them effectively can significantly reduce their impact. Our guide to Effective Incident Response [1] covers the processes, tools and skills that are likely to be needed; this document covers three basic measures that need to be in place on every computer if incidents involving it are to be handled successfully.

Every computer should keep some sort of record of events, since without this it will be impossible to discover what has happened and may be very difficult even to detect security incidents. Unfortunately logging is not a case of "the more the better" - logging everything is more likely to conceal the things that are significant and may even overload the computer. The right level of logging depends on what the computer is used for, but at least a record of login attempts (both successful and unsuccessful), reboots and system processes starting and stopping should be kept. For some services a log of requests to the service (again both those that are successful and those that are not) will be appropriate; for particularly critical systems it may be appropriate to log when programs or other critical files change or are accessed. Log files are likely to contain information that could affect the security of the computer, some will also contain personal data, so they need to be kept secure and protected from unauthorised access. Logs may be kept locally and/or remotely on dedicated logging servers: the latter can reduce the risk of an intruder altering the log to conceal his activities, but may increase the risk that the intruder will be able to interfere with the network communications and prevent log entries being written in the first place. Whatever approach is chosen, system and network administrators should practise using their logs both to develop skills and tools that will be needed in an incident and to discover before an incident whether the right information is being recorded.

Logs and other information on the computer will almost always be associated with the time when it was created. These times are normally taken from the system clock, so it is important that that clock is correct. Especially when trying to compare records from different systems (for example from a compromised computer, the router connecting it to the network and the security camera monitoring the room) if the different clocks are not synchronised it may be almost impossible to determine the true sequence of events. Organisations should ensure that their computers are synchronised to a common source of time; Janet provides a time synchronisation service [2] that allows organisations’ local time sources to be synchronised to global time standards.

Current attacks are so complex that the best way to recover a compromised computer after it has been investigated is to simply wipe the disk and re-install it from scratch. Taking regular backups of any data that might need to be recovered after such a re-installation is therefore vital. Backups or system images will also be needed for the restoration process: these need to be prepared carefully to ensure that they are secure and don’t contain either the tools that the intruder installed or the system vulnerability that allowed him to compromise the system in the first place. Compromises should be investigated so their root cause can be established and remedied. For example if the vulnerability was a weak password or a user clicking on a malicious link then even re-installing the system will not prevent it being compromised again.

Finally, detecting and responding to incidents will involve staff and processes from across the organisation. Individual computers and those responsible for them therefore need to be included in incident response and detection procedures. The minimum requirement is (as mandated by the Janet Security Policy [3]) to act when informed that their computer is a threat to the Janet network, however incidents will be less severe if they are also included in the technical and human networks used by their organisation for preparation, detection and response.

3.2 Security Measures on Client Computers

Client computers are exposed to security threats by three different routes - those targeting the user (for example invitations to download malicious documents or programs), those targeting the computer’s network connection (for example attacks on vulnerabilities in the operating system) and those arising from physical access to the device (if it is left unattended, lost or stolen). Technology and configuration can help defend against all of these threats, however security ultimately depends on user behaviour in taking reasonable care. Careless or deliberate behaviour by a user can defeat all technical measures.

Client computers cover a wide range of sizes and capabilities, from desktop workstations to laptops, tablets and smartphones. Although technical security measures exist for all these devices they may be less mature or less capable, for example the processing power on a smartphone may not be sufficient to encrypt and decrypt the whole of memory. For these devices safe user behaviour is even more important to make up for the limitations of the technology.

For computers fixed in one physical location, such as servers and desktops, it may be reasonable to address security by a combination of measures on the network, the device and the physical surroundings, knowing that these will always be present. However most client computers are portable and will connect to different networks at different times of day: in the office, at home, on hotel or conference wifi, or via mobile data services. Each of those networks will provide a different level of security protection: some will provide none. Mobile devices in particular therefore need to have as complete a set of self-defence measures as possible rather than relying on the network to protect them. The following sections describe the basic tools that are available.

3.2.1 Anti-virus

Over the past decade attacks on computers have changed: rather than breaking into the operating system to gain access to a computer, attackers are more likely to ask a legitimate user to run a program for them. Sometimes the program will be hidden in a webpage, document, spreadsheet or presentation - all of these have built-in programming languages that may run automatically - or infect a USB stick, or sometimes a program will be advertised as having a useful function but also provide the attacker with access to the computer when it is run. A number of different, and not always clearly defined, terms are used for different varieties of malicious program, though they are generally all considered different forms of malware (for malicious software).

Clearly user awareness is an important part of defending against malware attacks, though malicious web content is particularly hard to defend against as it may appear on otherwise legitimate sites whose comment or query interfaces are not very carefully designed. Disabling scripting languages and functions in browsers and applications may help, though it can also hinder legitimate use: many websites rely on scripts and many e-mails are unreadable if displayed in plaintext. The most effective technical defence remains anti-virus programs: even though they cannot detect all malicious code (and they are frequently criticised for this) they are still much better than nothing in almost all circumstances.

It is safest to assume that malware exists, or will be written, for every platform and operating system, so some form of anti-malware protection should be used on every client computer. Detection of malware may be based on known patterns in the program code or on particular kinds of unusual behaviour. Relatively low-powered devices such as smartphones may just scan new programs before they are installed; laptops and desktops are also able to monitor the activity of programs as they run. Both types of detection require frequent updates of the patterns they are searching for: devices should be configured at least to automatically check and notify when updates are available and normally, provided sufficient bandwidth is available, to automatically download and update them too.

Other forms of malware detection exist in web browsers and search engines: these can often be updated even more rapidly than device-based software so can warn of newly-infectious websites. However they will normally only provide a warning to the user, rather than actually blocking access to a malicious page, so rely on users recognising and respecting the warnings. Users must know not to ignore or disable warnings from the tools that are there to help them.

3.2.2 Host Firewalls

Firewalls are widely used in networks to protect computers against attacks, however mobile computers in particular cannot rely on always being behind such a firewall. Even if there is a firewall, if there is a malicious user or program behind the same firewall then the network firewall’s protection is little help. Operating systems for client computers now commonly include simple packet filtering functions that can significantly reduce the risk of a successful attack from the network. Since client computers should provide few services to the network, a default-deny policy for incoming connections is almost always a good idea.

More complex software firewalls are also available. Since these run on the client computer they are not limited to the sorts of packet-based filtering used by network firewalls, but can also control individual processes running on the same machine. A software firewall can check, for example, whether a particular program or operating system component should be permitted to make or accept connections to other computers across the network. This type of firewall is often provided as part of a suite of tools alongside more traditional anti-virus software. Such controls are potentially very powerful both in preventing attacks and in reducing their impact: even if malware succeeds in installing itself, it still needs to establish a network connection to its controller. A firewall that is aware of processes (assuming it has not been disabled by the malware) may be able to detect and prevent this and provide an alert of the problem. Unfortunately most users do not need to know what software is legitimately running on their computer or communicating with the network so designing suitable interfaces both for configuring these advanced firewalls and reporting alerts can still be challenging. Future improvements in these interfaces, and greater use of central configuration and management, are likely to make them even more useful than they already are.

3.2.3 Protecting Against Loss

Many client computers are now portable, for devices such as smart phones this is their main purpose. Such devices may be taken out of the office as a matter of routine so cannot rely on the presence of walls, locked doors, etc. to protect them. Furthermore many of these devices are both small, making them easy to lose, and high in value, making them an attractive target for thieves. Security measures for them should therefore assume that they will at some point come into the physical possession of a malicious person.

In such circumstances the main concern may well not be the loss of the physical device (which can be addressed by insurance), but the risk of unauthorised access to personal or confidential information either on the computer itself or to which it has access. Some protection can be obtained by requiring a passphrase or other secret to unlock the computer - this may well be the only protection if the computer is switched on when it is lost - but, since it will often be possible to defeat this measure by moving disks or other storage media to another computer, encryption of storage devices should also be considered. Provided the passphrase used to encrypt the storage cannot be guessed, this should prevent unauthorised access to the information it contains. High-quality encryption is now part of modern laptop operating systems and both free and commercial encryption tools are also available from third parties. Commercial programs may provide better options for central management though where laptops are only used to store working copies of information, rather than the originals, features such as key recovery may be less important. Many encryption tools can decrypt as part of the normal login process, so there is no need for the user to remember or enter an additional decryption passphrase so long as the one they use to log in is sufficiently hard to guess. Encryption should also be used for portable storage devices such as memory sticks if these may contain personal or confidential information.

Full device encryption is not yet common on PDAs and smartphones since, unlike laptops, these may not have sufficient CPU power to decrypt without a noticeable effect on performance. To reduce the risk to information, policies may be needed to limit the amount of information stored on unencrypted devices, for example by accessing information remotely rather than storing it locally and reducing the extent of automatic caching. The remaining risks may also be mitigated by using remote wipe features (typically triggered by sending a coded text message to the device) to delete information and by enabling remote location to increase the chance of recovering the device itself. Both tools need to be covered by policies and the implications discussed with users since remote wipe may also delete information the user considers their own and remote location could (but should not) be used to track the user’s location, considered by law as a serious breach of privacy. The need for such policies and discussion is even greater when users’ personal computers may be used to access organisational systems such as e-mail or calendar (known as Bring Your Own Device). The same technical measures are likely to be required to protect the organisation's information but only an agreed policy can address the interaction between the company's and the employee's reasonable expectations of how the computer and information will be used.

3.2.4 Human Behaviour

Probably the most important component of any computer security system is the humans who use and operate it. As noted above, most attacks now try to exploit vulnerabilities in humans rather than in computers. Furthermore computing power and storage are now so cheap that the impact of a human error can be catastrophic. In 2007 25 million people's bank details were lost on two CD-ROMs: in 2012 a single, even more easily lost, memory card could be used to store 1Kilobyte of data about each UK resident! If someone thinks it is a good idea to do this, then technology alone is very unlikely to be able to stop them. Any plan that hopes to improve security must therefore include helping humans to use computers more safely.

The good news is that surveys consistently show that most data losses and security incidents are the result of mistakes or mis-guided acts, not of malice. So raising awareness of when it is important not to make a mistake and getting better at pointing out the right way to do things - as well as designing systems that make the right way more obvious and reduce the likelihood of mistakes - could significantly reduce the number of such incidents. Furthermore safe behaviour online is no longer something that is just needed in the office: learning how to recognise and (not) respond to a phishing e-mail protects both your work webmail password and your home banking credentials; knowing the limits of what to share on a social network prevents both personal and professional embarrassment or worse. Some organisations benefit from this shared interest in safe behaviour by running sessions on how to surf safely at home, and then suggesting the same care be taken in the office. The most important principles aren't new, but just the transfer to the on-line world of things we were all taught about the real world as children: take care of information, don't believe everything/everyone you read, check if something seems suspicious.

But any efforts to improve behaviour will be damaged if the organisation's people, processes and systems either demonstrate or require bad behaviour. System administrators who ask users to disclose their passwords can't be surprised if they then fall for phishing attacks; managers who require staff to work from home but don't provide appropriate facilities can't be surprised if those staff log in to sensitive services from a family computer whose information and malware are shared with all the inhabitants of an on-line fantasy world. Good behaviour and good systems have to be established together.

3.3 Security Measures on Server Computers

3.3.1 Web Servers

Web servers are high-profile targets for attacks for at least three reasons:

- they are highly visible so an attacker who can publicly deface a website or take it offline can both damage the organisation's reputation and enhance their own (or that of their cause);
- websites that provide dynamic information often extract this from internal corporate

systems; an intruder who can control the web server may be able to extract other information that was not intended to be public from those systems or from the web server itself;

- popular websites have many visitors, so someone who can insert malicious code or instructions into the content served by the website can launch an indirect attack on all those visitors. Such attacks may either target vulnerabilities in users' browsers (known as 'drive-by malware') or lack of security awareness in users, for example by encouraging them to download malicious programs ('trojan horses') or to enter passwords or credit card information into the site ('phishing').

Web servers may therefore be used to attack both the organisations that run them and all those who may visit their websites.

The security of a web server is largely dependent on how well it is configured and operated because, unlike some services, there is not much that can be done through network configuration to protect it. The whole purpose of a public website is to be visible from (and therefore also open to attack from) the public Internet. Public web servers are likely to have to accept connections to their HTTP and HTTPS ports from anywhere on the Internet. Firewalls should be used to protect the other ports on the server; to make this easier it is generally not a good idea to run any other services on the web server but to run it as a single-purpose system.

Web servers can be attacked, so need to be secured, at three different levels: the operating system, the web server and the applications that generate dynamic content. Good practice guidance for securing all of these is available and should be followed. Log files from all three levels should be monitored for signs of attacks or reconnaissance. Particular care is needed when designing and using the interfaces for site administrators - if one of these privileged accounts can be compromised then the attacker may well gain complete control over what the site publishes and requests it receives as well as access to any internal resources that the web server uses to generate its content or process transactions. Administrative access should use encrypted network protocols wherever possible; two-factor authentication rather than simple username and password may be considered; some sites reduce the opportunities for attack by uploading content to an internal staging server from where it is copied by a scheduled update process running on the main server.

A common cause of problems on web servers is failing to check the input provided by users before processing it. It is easy to imagine that user input will only contain requests to view particular web pages, phrases to be entered into a search engine or text to be displayed as a comment on a blog, however all of these can be abused. Anyone writing code for a web server must remember that it will be run on the server by unknown and possibly hostile users, and take appropriate care in designing, implementing and testing it. The simplest form of attack, and one common to many Internet services, is simply to provide more input than the server expects. Limiting an HTML form field to display no more than dozen characters does not prevent a request containing 1025 characters being submitted. Systems and applications that handle user input must detect such over-sized entries and handle them gracefully otherwise they may provide a way to compromise the application, server or operating system. Web sites that use databases to generate content in response to user requests, perhaps just to search an archive of articles, must guard against requests including database commands known, from the most common query language, as SQL injection attacks. Similar attacks have also been used in the past to inject operating system or programming language commands where the web site used external programs to interpret those. Finally input checking is the only way to detect attacks against the website’s visitors, for example by including malicious HTML links or scripts in articles or comments, or uploading documents or executable programs that may exploit security weaknesses in visitors or their systems. Sometimes known as ‘cross-site scripting’, these attacks use both the site’s popularity to attack multiple users and its reputation to persuade those users that it is safe to click on links or download information from a trusted source. When sanitising user input to remove these problems it is better to aim to recognise content that is known to be safe, rather than trying to detect all possible variants of unsafe input. Systems to do this are unlikely to be perfect, but can significantly reduce the risk of a successful attack.

3.3.2 E-mail Servers

One of the most common problems with e-mail servers is computers running an e-mail service when they don’t need to. If a computer only processes e-mail under the direct instructions of a user running a mail client program such as Outlook or Thunderbird then it doesn’t need to run a mail server and any service that may have been started by default should be disabled permanently. These unnecessary mail servers are also likely to be old software versions with known security bugs so represent an unnecessary opportunity for successful attacks.

A common problem in setting up mail servers, both legitimate and not, is uncontrolled relaying. Most servers are intended to transfer mail between an organisation and the Internet, so for any valid message either the sender or the recipient, or both, should be a local system within the organisation. However some servers are prepared to accept messages from external sources for external destinations. This behaviour is known as relaying. While there are good reasons for allowing some servers to relay - to provide backup for another organisation or to support mobile users for example - the behaviour needs to be controlled by careful configuration. Open relays, which allow any combination of origin and destination, are frequently abused by advertisers and others to distribute bulk e-mail. This will usually overload the server, affecting its ability to handle legitimate mail, and often leaves the organisation with a flood of complaints and error messages to deal with. Sites that are frequently abused as relays may be added to blacklists used by many network operators and ISPs (Internet Service Providers) to reject all e-mail and other traffic.

Mail servers that offer a webmail interface are often used to generate unwanted bulk mail when legitimate users have been tricked into disclosing their passwords, often by a forged e-mail asking them to log in to something that looks like their university website or webmail service but is actually a fake created by the phishers. Webmail accounts compromised in this way can then be accessed by automated scripts to send mail from the tricked user, which is a particular problem when the recipients, trusting the sender, either click on a malicious attachment or are themselves directed to a phishing page. Users who disclose their passwords must change them, both on the webmail system and any other service (not forgetting those used for leisure) where the same password has been used. Not all e-mails apparently from a trusted person are the result of phishing: it is also possible to forge the origin of an e-mail without going anywhere near the genuine user or their mail system. This second type of attack doesn’t represent a failure of security by the person or service, though it may create significant amounts of error messages and complaints to the organisation. Unfortunately there is very little that can be done to prevent it only, as described in our guide to reliable mail systems, to design mail systems to be able to cope with these unexpected floods of traffic.

3.3.3 Domain Name Servers

The Domain Name Service (DNS) has a critical role in making the Internet usable by humans. Internet-connected computers use 32- or 128-bit numbers (IP addresses) to locate and identify each other but humans expect to be able to use names like www.example.ac.uk [4]. The DNS is the distributed directory that converts names into the correct numbers, as well as additional functions such as telling e-mail systems where messages for *user@example.ac.uk* [5] should be sent. Clearly it is important that DNS systems return the right answers to such questions: if they don't then a request for one website could return a completely different one or e-mails could be read or intercepted in a different part of the Internet.

Although it is common to refer to DNS as a single service, in fact there are two separate DNS functions that an organisation needs to provide. The first is to publish information about its own domain (e.g. example.ac.uk) to the rest of the world. Each service that will be accessed from outside the organisation (www.example.ac.uk [4], mail.example.ac.uk, etc.) needs to have a published record linking the domain name to its associated IP address(es). Second is to allow its own internal users to look up external domain names, a function known as DNS resolution. It is now considered good practice to separate these two functions and to provide them on separate systems, not least so that an increase in requests to the DNS server from outside the organisation doesn't cause name resolution to become slow or fail for those inside. The DNS protocols include facilities to replicate both DNS server and DNS resolver functions across a number of different computers or even across different physical locations to increase the reliability of this critical service.

Someone with control of the organisation's authoritative name server can effectively replace any of the organisation's public servers with another computer of their choice, anywhere on the Internet. Attackers often use this to cause public embarrassment by having an organisation's www name resolve to another website publishing critical comments about them, but much more dangerous attacks are also possible. Systems providing name server functions must be maintained and managed in accordance with best practices for both host and network security. A service that needs particular care is dynamic DNS, which allows individual computers to update their own information in the authoritative server; clearly it is important to use careful configuration and authentication to ensure that such updates will only be accepted if they come from the expected origin. Where a DNS service does not require dynamic updates these should be disabled. Since DNS is a distributed database it is also vital to protect the links between different parts of the database, in particular the information that says which machines are the name servers for your *example.ac.uk* domain. For names within .ac.uk this information is maintained and published by Janet; for other domains it will be maintained by whatever registrar allocated the domain name. Registrars offer a number of different ways to authenticate requests to change this nameserver information: whichever you choose, make sure that it is hard for an unauthorised person to forge, since doing so will transfer your entire Internet namespace to their, rather than your, control.

While an attack on a DNS name server mainly affects people trying to contact the organisation from outside, an attack on a DNS resolver generally affects those inside the organisation whose DNS requests will produce the wrong answer. Like a DNS server attack this can be used to embarass the organisation, but a more serious risk is that it can be used to reveal confidential information. By inserting false information into the DNS resolver an attacker could, for example, redirect requests for the organisation's webmail or intranet server to a machine they control in order to harvest the usernames and passwords that are entered into it. Attacks on the DNS resolution process have also been used to gather e-banking and credit card details by re-directing requests for sites where those are entered. For a long time this has been done by poisoning attacks: providing false information to a DNS resolver that it will then serve to clients that request it. When a client asks its DNS resolver to look up a particular name, the resolver will itself make requests to other servers to find the answer; some resolvers will accept all the information contained in a response, even if it is irrelevant to the original request. A malicious server that receives a request can use this to insert false information about other domains: either the organisation's own or the domain of a target bank or e-commerce site. More recently it was discovered that a flaw in the DNS protocol could allow any well-connected system on the internet to insert poisoned information into a resolver, even if the attacking machine was not itself part of the DNS hierarchy. Most DNS software now includes options to make both attacks more difficult; these should be used whenever available. DNS resolvers should be configured to only accept requests from the organisation(s) they are intended to serve, not from the whole Internet: this makes poisoning attacks harder but, perhaps more importantly, limits the likelihood of the resolver being used as part of a denial of service attack against a third party.

The long-term solution to attacks on DNS resolution is a system called DNSSEC. In addition to publishing translations between internet names and addresses, DNSSEC also publishes cryptographic information that can be used to check that those translations originate from the organisation they should. Provided the systems that perform DNS resolution also validate the responses they receive, invalid information will be detected and rejected. DNSSEC needs to be implemented on both authoritative servers (so that others can check for forgery of the organisation's own domain) and on resolvers (so they can check for forgery of other domains).

3.3.4 Telephony Servers

A new challenge in securing servers is the increasing links between IP networks and telephony. Connecting a telephone switch to the Internet connects two different security models - unlike the Internet, telephone networks do include some central security measures - which means these devices have to implement both Internet security good practice **and** telephony security good practice. Unlike most Internet services, a security breach on a telephone switch can directly cost the organisation money: attackers commonly use compromised switches to make international phone calls (either for themselves or their paying customers), or to dial premium rate numbers from which they share the revenue. In both cases the cost will appear immediately on the organisation's phone bill. Like any other system, attacks can exploit weaknesses in the user, their client, the server, or the communication protocols. Security measures are likely to be needed to protect all of these.

The main user-related problem actually affects all telephone switches even if they are not connected to an IP network - default or blank passwords. Many switches allow users to access their voicemail boxes remotely; some also allow users to make calls through the switch or to set up call forwarding. Clearly if an attacker can obtain the password for voicemail access (or if there is no password set) there is likely to be a breach of confidentiality; if the same password allows the attacker to make or forward calls then they can do this at the organisation's expense. All accounts should either have non-default passwords set or else remote access removed. Even where remote access is needed, organisations should consider what functions should be enabled for remote users. It may also be possible to check for excessive calls (or bills) either on the switch at the point where calls join the telephone network.

When connecting either a telephone switch or a telephone to an IP network, it is important to remember that they are both network-connected computers and their security needs to be managed accordingly. The basic measures recommended elsewhere in this chapter for clients and servers are needed for telephony systems too. Network controls discussed in the following chapter should also be considered - definitely for any administrator interfaces and if possible for user interfaces too. If local users need to connect when offsite it may be safer to do this over a Virtual Private Network (see the next section on Remote Login) rather than allowing direct user registration to a telephony server from anywhere on the Internet. Registration should at least require the users to authenticate themselves to their telephony server.

The protocols used to carry telephony traffic over IP are complex and may require special handling at firewalls and network address translation devices. In particular note that the signalling and audio/video traffic may not follow the same paths through the network: signalling will generally be routed via switches, audio/video may try to go direct between clients. This means that a dedicated gateway at the edge of the organisation’s network may be needed if IP telephony is to be used across the Internet or to/from networks using NAT. IP telephony protocols often use UDP rather than TCP, which increases the opportunity for packets to be forged. An attacker who is between the client and server (or who is able to persuade the client and server to route traffic via them) may be able to listen in to calls, insert content, or forge signals to close or overload the call. These problems arise wherever IP telephony traffic passes over an untrusted network and are particularly acute if wireless networks are used to connect IP telephones. Carrying telephony traffic in an encrypted tunnel may be the only way to prevent attackers from listening, hi-jacking or disrupting calls.

3.3.5 Remote Login

Many networked services involve the user logging in, often by providing a username and password. Transmitting some sort of secret across the network is likely to be an early stage in everything from accessing a social network or e-mail service to remotely controlling a server or experimental equipment. Information that is sent or received after logging in may also be sensitive, for example if it contains personal or financial information or configuration details that could be used to compromise security. Many of the original internet protocols did nothing to protect this information, assuming that all those with access to the network were trustworthy. With the great increase in use of and access to the Internet, and the use of technologies such as wireless that are much easier to "listen in" to, that assumption is no longer sensible. Both services and user computers should take precautions to protect any sensitive information that they send or receive across the network.

Many of the early, unencrypted, protocols now have encrypted versions that do protect information between the client and server. Secure HTTP (web), IMAP and POP (e-mail) are widely supported; SSH is an encrypted replacement for telnet for command line access. Services that communicate via these protocols should be designed to use the secure versions at least for logging in. If all clients can be relied on to support the encrypted versions then it may even be possible to turn the unencrypted ones off or at least block direct access to them from the Internet to prevent users exposing their login credentials to the greatest risk.

Where unencrypted protocols still have to be supported this can be done by first establishing a secure tunnel across the untrusted network, either to the service itself or, if parts of the organisation's internal network can be trusted, to a dedicated device within the organisation that will then pass the decrypted connection on over the trusted network to the service. A number of different encrypted protocols can be used to establish such Virtual Private Networks (VPNs), including SSL, SSH and IPSec: the choice is likely to depend on the range of clients that need to be supported and the locations from which they need to connect (networks in hotels, conference centres, etc. may be more likely to permit SSL, but commercial organisations may be more likely to block or intercept it). Where a dedicated server is used as the organisational endpoint, this VPN concentrator can implement additional security measures such as requiring authorised users to log in before handing on their connections. Internal servers can then be configured to only accept insecure versions of protocols such as IMAP or NetBIOS if they come from the concentrator's IP address(es) rather than leaving them open to the whole Internet. Note, however, that a VPN can also provide an effective route around the firewall or other measures such as malware scanning that may be implemented at the connection between the organisation and the external network. Since the whole point of the VPN is to conceal the content of the traffic it carries, firewalls too will be unable to inspect it. The interaction between VPNs and other network control measures needs to be carefully planned and the implications of permitted traffic flows understood.

VPN concentrators are particularly useful when systems need to be administered remotely - for example using telnet, VNC or RDP - if the administration interface does support an encrypted protocol. Rather than leaving the administration port open to the whole Internet, exposed to both software vulnerabilities and password-guessing attacks, users first need to prove their authorisation to the VPN concentrator - a machine that can be configured and maintained to the highest security standards. This should both reduce the opportunities for attacks and ensure there is a clear audit trail if one occurs.

Even if it can be carried in an encrypted tunnel, remote administration of some systems may be too much of a risk if it is only protected by a static password that an unauthorised person can capture and reuse to gain access later. For these systems it may be appropriate to consider two-factor authentication mechanisms where additional information is required to authenticate a user: perhaps a certificate, one time code from a keyfob, or biometric information such as a fingerprint.

Source URL: https://community.jisc.ac.uk/library/janet-services-documentation/computer-security

Links
[1] https://community.ja.net/library/janet-services-documentation/effective-incident-response
[2] https://community.ja.net/library/janet-services-documentation/network-time-service
[3] https://community.ja.net/library/janet-policies/security-policy
[4] http://www.example.ac.uk
[5] mailto:user@example.ac.uk