Basic computer security strategies and models

Ishbel Duncan ishbel@cs.st-and.ac.uk
School of Computer Science, University of St Andrews

Overview of commonly used security strategies and models. However, these are only as good as the people enforcing them and the people using them.

Although in the internet world we still store paper copies of personal data, our work and personal computers contain details of both our personal and our corporate lives. We transmit Megabytes of data in our emails and are bombarded daily with spoof emails, spam or 'friendly fire' emails with embedded pictures or graphics. Whether we use a work PC or our own personal PC/laptop, we cannot escape the quantity of data flowing through the internet into and out of our machines; business email alone last year was estimated at around one billion gigabytes. Our lives, and those of others, can be elicited from our files, our emails and our pictures. If we log on to the internet we open ourselves up to invasion of privacy; if we lose a data CD, we hand over both explicit and implicit information to the person who finds it. To lose a laptop, or even a posted data CD, causes grief not only to the personal owner, but possibly also to the corporation. We protect our wallets and our handbags by keeping them on our person or locking them into drawers at work, but we leave our PCs running, leaving information open to abuse. If we have an internet connection and do not protect our files, it doesn't take a computer genius to attack and scan for personal or financial data or even just delete files out of malice. Ultimately, we need to protect both our systems and our data as a corporation and as an individual.

Data security has a long history from ancient times, including the famous Caesar cipher, through to modern data cryptographic protocols based on keys (data patterns) of 512 binary digits or more. Computer security models and strategies came more into use with the advent of the multi-user systems in the 1960s when users required identifiers and passwords and were allocated some file storage space in the one and only computer a company, university or medical research establishment would own. Data files were protected separately under the UNIX file systems of the early 1970s by marking files as being readable, writeable (updateable) or executable (runnable) by the owner, a designated group of users, or the world (meaning everyone else on the system). Variants of this strategy are still used today and access rights underlie one of the commonest strategies used – the **Access Control List**. For any large system, a systems administrator can set up a matrix of users against objects, where objects can be, for example, files or database tables. Essentially, each object has a list of users and their associated access rights. A separate way of distinguishing access is through **Role Based Access**. For example, in a GP surgery one would presume the GPs themselves would have access to all patient records and drug information. The administrative staff would have access to some parts of the patient record, such as name and address, and schedule information. They should not be able to see the drug list associated with an individual patient. Effectively, the access control list for the GP would show read-and-write access to a patient's file, and read access to the drug list. The administrator would have read and write access to the patient's details, but not the whole file, which is essentially split into two parts to allow separate access to the details and to the patient history. Thus users, or groups of users, have access to different parts of the system or data. Problems, or weaknesses in the system, may arise when one user has dual roles or a superuser is given rights to all parts of the system.

Data security has a long history from ancient times, including the famous Caesar cipher, through to modern data cryptographic protocols based on keys (data patterns) of 512 binary digits or more.

So, we can protect files by a series of read-and-write privileges but we can also take the security back to the user level by incorporating **user authentication**. Identifiers and passwords are common but are invariably easy to attack given that over 60% of users tend to utilise simple dictionary or proper name based passwords. Allegedly another, hopefully mostly separate, 60% use the same password for more than one system access such as their work PC, Hotmail, Amazon and online banking. A simple form of attack is to use a word generator which can work through the average length password of 8 characters in a matter of minutes. Given that many people use private family or pet names, by simply

Security can never be presumed and can always be compromised.

looking at the pictures on their desk or office wall, an internal attacker may elicit enough incidental information to reduce the attack time. We are, as a nation, very careless with our privacy.

We have to take responsibility for how accessible we leave our machines, how often we install yet more security protection software and how simple we make our passwords. A basic policy must be *semper vigilo*.

Apart from protecting files and user access, protection can be placed at the network level. **Firewalls** may be placed to shut out the vast majority of troublesome emails and viruses and may also be used to ensure that only valid users of a system have access to the internal computers that make up the local network. A firewall can be a dedicated machine, a processor, scanning incoming or outgoing data for abusive behaviour or it can be a program running on your PC or laptop scanning for malware such as viruses. Networks can also be configured to only allow data transmission between two trusted parties, via **Public Key Infrastructure** (PKI) certificates such as you get when making an online financial transaction. PKI uses encryption, with a public key used by the sender to encrypt the message, and a private key used by the recipient to decrypt the message. A trusted authority issues the digital certificate which includes the public key information.

There are many layers to each level of security but each level (the user authentication, data access and network access) is still subject to human flaws, attacks and, of course, failures.

Many companies employ a human-centric security policy as another way of controlling how people interact within a system. A computer security model specifies and enforces a security policy, expresses what the protection mechanisms must achieve and may outline data access conditions, user validation procedures and possible threats.

The basic security model comes, not surprisingly, from the military and is based on the need-to-know principle. In the **Military Model**, access is given through a linear hierarchy from unclassified, general access, through restricted and confidential, up to secret and top secret. How these layers of secrecy are enforced is not part of the model, but this model is relevant to both industry and research data and systems.

The **Bell–LaPadula** security model identifies allowable communication while maintaining secrecy. Information cannot flow downwards in a hierarchy so no-one, or no process, can write down to a lower level in the hierarchy. No-one, or no process, can read data stored at a higher level. Thus sensitive data can only be written to the same or a higher level. A CEO can read files belonging to the staff but they can only write information up to their line managers and so on up to the CEO.

The **Harrison Ruzzo Ullman** security model uses access control matrices as the basis of system access and control. Objects in the system, users or files, can grant access rights such as owning, reading, writing and propagating. The model is based on commands where each command involves conditions and primitives such as 'if level = nurse then allow access' and essentially reads like a computer program.

The **Chinese Wall** security model reflects protection requirements for commercial or legal information, and is a way of avoiding conflicts of interest. Here, all objects (files) pertaining to one company are grouped together in a named group. Conflict classes are created where all groups of objects for competing companies are clustered together. For example, a conflict class would contain say Shell, Esso and BP files, each within distinct groups. A person can access any information as long as they have not accessed information from a different company in the same conflict class. A person accessing Esso files could not then access Shell files because they are in the same conflict class. They could, however, access Tesco files as they would be in a separate conflict class containing Asda, Morrisons and Sainsburys.

The **BMA** model was developed in 1995 by Prof. Ross Anderson of Cambridge University. This was based on access control lists where a clinician must be named to open the file belonging to a patient. Both a referring GP and a surgeon would need to be named on a patient file for a referral of a patient to a hospital. One of the clinicians would maintain control of the file. All accesses to that file would be logged and no-one would have the right to delete clinical information. Patients had to be informed if their files were to be aggregated for clinical studies and measures had to be taken to anonymise data.

All of these models or strategies are only as good as the people enforcing them through computational safeguards such as efficient and updated access control lists, strong passwords and ever-vigilant network controls. Each of these systems can break down over time by secondary changes made with the best of intentions. A simple access or role change can have a ripple effect across a security system and it is important to note that, once a system or model is introduced, it has to be constantly enforced and checked after every user or file-access change. Backdoors into systems are a common form of attack caused by failure to note that a single access change may affect a whole subsystem. A quick and dirty fix one day can leave a hole in the security wall. Technical security measures can never be totally secure: encryption can be broken; staff within an organisation can subvert the system; staff in the technology production company can introduce back doors. And, as we have shown above, these security measures can also be compromised by users themselves through lax use of passwords and PCs and cavalier use of portable laptops and data storage media.

Security can never be presumed and can always be compromised. We have to take responsibility for how accessible we leave our machines, how often we install yet more security protection software and how simple we make our passwords. A basic policy must be *semper vigilo*.

RESOURCES

Wikipedia entry: Computer security model. <http://en.wikipedia.org/wiki/Computer_security_model>.

Anderson R. Security engineering. (e-book) 2001; <www.cl.cam.ac.uk/~rja14/book.html>.