# Global Information Assurance Certification Paper

Jerry Marsh
January 2001

# Myths Managers Believe About Security

Managers, particularly upper level and non-Information Technology related managers, hold a number of beliefs about security that are not true. These beliefs hamper effective security implementations at many organizations. Following are some of those myths.

**MYTH: Our firewall product protects us from the Internet**

Typically, this myth is false in a number of different ways.

First, a firewall is a perimeter defense mechanism. For it to work effectively, it must be the only path to the outside world[1]. This is generally not true. First there are dialup accesses, both known and unknown, into the network. Additionally there are all too often connections to "trusted" third party networks. All of these create potential "back door" paths to the internal network.

Second, a firewall is only as good as its rule base. A permissive rule set ("permit any" in both directions being the extreme case) makes a firewall into an expensive router. Most organizations create a rule set that blocks most traffic originating from the Internet (inbound traffic), and permit most or all traffic originating from the inside network to the Internet (outbound traffic). This creates two issues. In many firewall implementations, the content of permitted inbound traffic is not checked by the firewall. (A few support a hand off of selected traffic to separate processes for virus checking - e.g., SMTP mail attachments, FTP file transfers). But who is to say that port 25 traffic (normally SMTP) is really email traffic, especially if it is not limited to reaching the known SMTP email machine or machines. How do we know that port 53 traffic is really a DNS request, or that what claims to be a PING (ICMP request/ICMP reply) is not really carrying an unexpected payload?

The larger issue, however, is the often unrestricted and often unwatched outbound traffic. A variety of possibilities exist for "back channel" connections from the inside out to the Internet. I know personally of one case where a contractor with legitimate dialup access did not like the speed of their connection, and thus used the dialup into a Unix machine to start an X-window session back via a high speed Internet connection to their business. (X-window sessions are typically made from the application server back to the X-terminal station.)

A recent "gray hat" product, HTTPtunnel[2], permits tunneling traffic through a firewall on port 80. While initial implementations of this protocol did not provide a generalized tunneling capability (i.e., the ability to have general access initiated from either direction once a tunnel is established), it is only a manner of time before this kind of functionality

will exist. And this is but one of several tunneling implementations available to thwart firewalls[3].

Another major problem is mainstream vendor's intentional attempts to circumvent firewall controls. Bruce Schneier, in his June 15 crypto-gram[4] talks about this issue with SOAP (Simple Object Access Protocol). Microsoft, in their own documentation[5] says (emphasis added):

*"**Firewall Woes***

*Currently, developers struggle to make their distributed applications work across the Internet when **firewalls get in the way**. Since most firewalls block all but a few ports, such as the standard HTTP port 80, all of today's distributed object protocols like DCOM suffer because they rely on dynamically assigned ports for remote method invocations. "*

***. . .***

*"Since SOAP relies on HTTP as the transport mechanism, and most firewalls allow HTTP to pass through, **you'll have no problem invoking SOAP endpoints from either side of a firewall**."*

***. . .***

*"Combining HTTP and XML into a single solution gives you a whole new level of interoperability. For example, lathered with SOAP, clients written in Microsoft® Visual Basic® can easily invoke CORBA services running on UNIX boxes, JavaScript clients can easily invoke code running on the mainframe, and Macintosh clients can start invoking Perl objects running on Linux. **The list goes on**. "*

Though I have quoted Microsoft here, this initiative is a collaborative development of several vendors. These developers are essentially saying that firewalls keep them from doing what they want to do, so they develop protocols to circumvent firewall controls.

Planned corporate Virtual Private Networks (VPNs) are another source of firewall bypass. Because of the network traffic load on firewalls, and the load on the VPN resulting from VPN encryption, organizations often place the VPN tunnel service on a different machine than the firewall service. Either the firewall has rules that permit unrestricted VPN tunnel traffic through firewall to the VPN server, or the VPN server is placed in parallel with the firewall. In ether case, the firewall provides no protection for the VPN traffic. The remote VPN client, appears as if they were directly attached to the internal network at the point that the VPN server is and often has unlimited network access. In this case, it is a logical "backdoor" connection. In reality it is coming through or by the "front door", but is not controlled by the firewall. Unless the VPN implements its own firewall like controls, there is no restriction of access once the tunnel is

established. Neither the firewall nor virus checkers can see into the packets to determine their content while they are in their tunneled encrypted form.

Lastly, it is possible that someone will develop a Trojan that intentionally does not do anything obviously "bad" to the host machine. But on command, perhaps through a Ping command, or a DNS "request" or "response", or an email, starts up a VPN tunnel from the inside out to an outside machine enabling an outside user to remote control the machine, or simply to use the machine as an IP gateway to the rest of the internal network.

A firewall is a perimeter tool, much like the guard posted at the entrance of a planned housing development. The guard of such a housing development keeps general traffic out of the development. This does not eliminate the homeowner's responsibility to store their valuables in a locked safe and lock the doors and windows of the house.

**MYTH: We haven't been broken into so far, so we must be doing a good job of security**

This myth is false in two ways.

First, how do you know you have not been broken into? Many organizations do not audit changes to their system software and applications. How would they know when something has been modified? In my previous employment, I have had calls from both FedCERT and the FBI telling me about customer servers that had been compromised. In both of these cases, the servers were outside any firewall environment. In the FedCERT case, a web page was obviously changed. In the FBI case, the changes were not obvious – in fact, they were on an "inactive" server.

Within the last two weeks of this writing, another customer public web server had a page modified. They corrected the page. Just today, I received an email complaining about an "ICMP flood" from that machine (my name is still associated with the address record for that machine, even though I have changed jobs). Apparently there is other damage on that machine besides the corrupted web page. Who knew that until I got the complaint? And who knows the extent of the compromise? Without good change management and system auditing processes in place, it is not possible to determine when and what software has been compromised[6].

Second, just because you have not been hit yet, doesn't mean your security is good. You may just be lucky.

**MYTH: Our Information Technology products provide for good security**

The "gotcha" in this myth is "provide". Many products have considerable support for security (none without room for improvement). However, vendors sell products better when they install as easy as possible. As a result, products come from the vendors with security features turned off or defaulted off. Out of box installs result in poor security

environments. It takes someone security knowledgeable and with a security strategy in mind to implement products well from a security perspective.

Examples:

The default NT server password hashing stores two hashes (for compatibility) of passwords. The older "LAN manager" hash, is notoriously easy to crack.

Cisco routers by default do not require logon authentication. Also by default, passwords are passed in plain text over the network.

By default with NT server, a "guest" account is active. Passwords for accounts are not required to be used, and if used are not required to be "strong" nor to be changed.

**MYTH: The Information Technology organization can manage the security issues**

This is not true in most organizations. There is a natural conflict between "ease of use/function" and "security". Unless the IT organization is powerful enough to decide and enforce security policy, ease of use/function will win.

There needs to be a "security organization" which is positioned high enough and broad enough in the corporate structure to be able to set policy and have it stick[7]. I once worked for a vendor at a company where it was grounds for dismissal if your password was shared with anyone. In six year's time, I never saw or heard anyone share their password. The appreciation for security is a corporate wide issue. The "security organization" needs to educate the corporation on the cost of not doing security. Often the IT organization is busy providing IT infrastructure. The IT organization will play a major role in implementing security policy and even help formulate it.

**MYTH: Technology products solve the security problem**

There is a belief that firewalls, VPNs, Intrusion Detection System (IDS) servers, and auditing products will create a secure environment. These products are just tools[8]. They are only a piece of the solution. An information technology vulnerability and risk assessment is another. A corporate security strategy is another. Security education is another. Manpower to review security alerts and security patch lists, implement security patches, implement security policy, monitor logs, and audit systems is another. A firewall or a server or an IDS server that logs information that no one looks at is of limited use. If no one audits servers for unauthorized changes, how will the organization know they have been compromised?

**MYTH: Our anti-virus scanner protects our computers**

There is some truth to this. For those known viruses that are in the anti-virus vendor's database, viruses will be detected. However, there are two significant problems with this. New viruses (i.e., those not yet seen by the anti-virus vendors) are not protected against.

It could be days to weeks before a new virus in detected, analyzed, and the signature file distributed to customers. During this period, computers are vulnerable[9].

Second, there are polymorphic viruses[10.] These viruses change appearance as they replicate. Since each replication of the virus looks different, it is difficult to produce a signature file to detect the virus in transit.

Anti-virus scanners should be used. However, users need to be educated NOT to trust them as absolute protection. Any suspicious attachments should not be opened. Automatic execution of system functions, such as email preview, should be turned off.

**MYTH: Our NIDS server will detect intrusions**

Like virus scanners, most Network Intrusion Detection Systems use a database of signature files to detect known exploits. If a new exploit is not in the database, it will not be detected. This is complicated because most exploits can have a variety of appearances as the traverse the network. Programs exist that fragment packets, send fragments out of sequence and overlapping to thwart NIDS servers. Some NIDS servers have methods of "normalizing" the traffic before checking[11], but these have varying degrees of success.

In a recent article, the use of Unicode and its impact on IDS detection is discussed[12]. A single exploit could exhibit one of literally millions of different forms as it travels across a network.

NIDS can only detect what they can see. If traffic is encrypted as it passes the NIDS probe, or if the NIDS probes are not deployed in the right places in the network, they can't help. The increasing use of switches rather than hubs tends to make NIDS probes blind.

Finally, many NIDS systems work by alerting someone when suspected exploits are happening. As was demonstrated at the October 2000 Monterey SANS conference, this can be thwarted by information overload. In this example the attacker created so many "noise" attack attempts that people watching for attacks were overloaded. The real attack was injected in the middle of the noise and completed before it could be determined what the real target was.

**MYTH: We don't do anything that makes us a target for attack**

The most obvious response to this myth, is "why do you need to be a specific target?" Some attackers don't need a reason. Another server compromised is yet another notch on the virtual gun.

There are a variety of other motivations for compromising computer systems.

One of the more recent concerns is political terrorism. The target for this threat is any organization, government or private, that is perceived to make a significant contribution

to the well being of society. Public safety, medical care, power, water, transportation, finance, and communications related organizations are all potential targets. Also, any organization that will, if compromised, make good media coverage for a political statement is a target.

Anger is also another motivation. A disgruntled employee or ex-employee might consider discrediting their employer by compromising their systems. Especially for government organization, public citizens unhappy with the way "government" is treating them have a motivation for "getting even".

"Industrial espionage" is another motivation. Normally this is thought of in terms of stealing a competitor's secrets for business advantage. However, any organization that handles, manages, or awards large amounts of money or contracts is also a target. If "inside information" can influence the disposition of large contracts or money transfers, then there is a large incentive to steal or modify information. In this case the attacker will work in ways to avoid detection. Changes to software and databases will purposefully NOT be obvious. This is perhaps the worse case of unwanted intrusion. It may be months or years before an organization is aware that the integrity of their data has been compromised, and they may not have a practical way of recovering the compromised data in their systems.

**SUMMARY**

A good comprehensive security posture is much more than security hardware and software components. There is no "silver bullet". Good security requires a security vulnerability assessment and risk assessment. The need for security must be recognized and supported across the organization. There must be a corporate wide security policy and structure. Security must be an integral part of infrastructure and application design. Changes to the environment must be managed. The security components in the infrastructure (servers, routers, firewalls, IDS servers, etc.) must be monitored and analyzed looking for unexpected changes.

"Security is not a product, it is a process."[8]

References:

[1] Ranum, Marcus and Curtin, Matt. "What can't a firewall protect against". Internet Firewalls: Frequently Asked Questions. Rev 10.0. 1 Dec 2000.
URL: http://pubweb.nfr.net/~mjr/pubs/fwfaq/ (22 Jan 2001).

[2] Brinkoff, Lars. Free Software Foundation/nocrew.org. (program source).
URL: http://www.gnu.org/software/httptunnel/httptunnel.html
URL: www.nocrew.org/software/httptunnel.html (22 Jan 2001).

[3] Rideau, François-René. "Firewall Piercing mini-HOWTO". Version 0.7. 7 Nov 2000. URL: http://www.fokus.gmd.de/linux/HOWTO/mini/html_single/Firewall-Piercing.html (22 Jan 2001).

[4] Schneier, Bruce. "SOAP". Crypto-Gram. 15 June 2000. URL: http://www.counterpane.com/crypto-gram-0006.html (22 Jan 2001).

[5] Skonnard, Aaron. "SOAP: The Simple Object Access Protocol". January 2000 URL: http://msdn.microsoft.com/library/periodic/period00/soap.htm (22 Jan 2001).

[6] Crabb, Michele. "Maintaining your Infrastructure". Chapter 5. Building a Successful Security Infrastructure. SANS 1998 Training Course Book.

[7] Crabb, Michele. "Management Commitment: Why is it so important?". Page 1-8. Building a Successful Security Infrastructure. SANS 1998 Training Course Book.

[8] Schneier, Bruce. "Security is not a product, it is a process". Crypto-Gram. 15 Dec 1999. URL: http://www.counterpane.com/crypto-gram-9912.html (22 Jan 2001).

[9] Russell, Ryan and Cunningham, Stace. "Viruses and Trojans Cannot be 100 percent Protected Against". Pages 41-44. Hack Proofing Your Network.

[10] Pakistan Computer Emergency ResponseTeam. ""Polymorphic Viruses". URL: http://pakcert.com.pk/pages/Polymorphic_Virus.htm (22 Jan 2001).

[11] Hoglund, Greg and Gary, Jon. "Multiple Levels of De-synchronization and other concerns with testing an IDS system". 11 Aug 2000. URL: http://www.securityfocus.com/focus/ids/articles/desynch.html (22 Jan 2001).

[12] Hacker, Eric. "IDS evasion with Unicode". 3 Jan 2000. URL: http://www.securityfocus.com/frames/?focus=ids&content=/focus/ids/articles/utf8.html (22 Jan 2001).

Author's note: As I finish revising this document, I receive word of yet another customer of my previous employer with a compromised machine (this is in addition to those quoted in the text). To be fair, this is a "public" machine (outside any firewall). So far, I have received over half a dozen complaints from sites that this compromised machine is scanning them.