



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Hacker Tools, Techniques, and Incident Handling (Security 504)"
at <http://www.giac.org/registration/gcih>

**"Attack vs. Defense
on an Organizational Scale"**

GCIH Gold Certification

Author: Omar Fink, omarfink@aol.com

Advisor: Don C. Weber

Accepted: November, 2007

Table of Contents

| | |
|--|----|
| 1 - Introduction..... | 4 |
| 2 - Preparation..... | 8 |
| A - DEFENDERS: Policy and Paperwork..... | 8 |
| 1. NIST Special Publications and Security Policy..... | 8 |
| 2. RA-3 RISK ASSESSMENT (defender's policy)..... | 13 |
| 3. PL-2 SYSTEM SECURITY PLAN..... | 14 |
| 4. Threat Analysis..... | 15 |
| 5. Scenario: Defenders' Policy..... | 17 |
| B - ATTACKERS:..... | 20 |
| 1. Col. Boyd's OODA Loops..... | 20 |
| 2. Situational Awareness Matrix..... | 22 |
| 3. SCENARIO - "The Academy"..... | 26 |
| 3 - Using Google for Reconnaissance..... | 30 |
| A. Scenario (Google)..... | 30 |
| 4 - Perimeter..... | 37 |
| A. DEFENDERS: SI-2 FLAW REMEDIATION..... | 37 |
| B. ATTACKERS: METASPLOIT..... | 38 |
| C. Scenario (Perimeter)..... | 43 |
| 5 - Wireless Network..... | 48 |
| A. DEFENDERS: AC-18 WIRELESS ACCESS RESTRICTIONS..... | 48 |
| B. ATTACKERS: KISMET..... | 50 |
| C. ATTACKERS: Aircrack..... | 50 |
| D. Scenario (Wireless Network)..... | 52 |
| 6 - Bypass..... | 64 |
| A. DEFENDERS: SI-3 MALICIOUS CODE PROTECTION..... | 64 |
| B. ATTACKERS: Custom Malware..... | 65 |
| C. Scenario (Bypass)..... | 67 |
| 7 - Walk-in..... | 69 |
| A. DEFENDERS: PE-3 PHYSICAL ACCESS CONTROL..... | 69 |
| B. DEFENDERS: AC-11 SESSION LOCK..... | 70 |
| C. DEFENDERS: AC-6 LEAST PRIVILEGE..... | 71 |
| D. Scenario (Walk-in)..... | 71 |
| 8 - Entrench..... | 77 |
| A. DEFENDERS: IA-2 USER IDENTIFICATION AND AUTHENTICATION..... | 77 |
| B. ATTACKERS: A Simple Batch File..... | 78 |
| C - ATTACKERS: Password Cracking..... | 83 |
| D - Scenario (Entrench and Crack)..... | 87 |
| 9 - Zero Day..... | 92 |

| | |
|--|-----|
| A. DEFENDERS: SI-4 INTRUSION DETECTION TOOLS AND TECHNIQUES..... | 92 |
| B - ATTACKERS: Zero Day Exploits | 93 |
| D. Scenario (Zero Day Attacks)..... | 96 |
| 10 - Distributed Denial of Service Attack..... | 98 |
| A - ATTACKERS: DDOS..... | 98 |
| B - Scenario (DDOS)..... | 99 |
| 11 - Aftermath and Lessons Learned..... | 103 |
| A. DEFENDERS: Effectiveness | 103 |
| RA-3 RISK ASSESSMENT..... | 103 |
| PL-2 SECURITY PLAN | 103 |
| AC-11/12 SESSION LIMITS/TERMINATION..... | 104 |
| AC-18 WIRELESS RESTRICTIONS..... | 104 |
| IA-2 USER IDENTIFICATION AND AUTHORIZATION..... | 104 |
| IA-3 DEVICE AUTHENTICATION | 105 |
| PE-3 PHYSICAL ACCESS CONTROL | 105 |
| SI-2 FLAW REMEDIATION | 105 |
| SI-3 MALICIOUS CODE..... | 106 |
| SI-4 INTRUSION DETECTION..... | 106 |
| B. DEFENDERS: Results..... | 107 |
| C. DEFENDERS: Lessons Learned..... | 108 |
| D. ATTACKERS: Effectiveness | 109 |
| Perimeter Attack..... | 109 |
| Wireless Attack..... | 110 |
| Bypass Attack..... | 110 |
| Walk-in Attack..... | 111 |
| E. ATTACKERS: Results..... | 111 |
| F. ATTACKERS: Lessons Learned..... | 111 |
| 12 - Concluding Notes | 113 |
| 13 - References..... | 114 |

1 - Introduction

Historically, the motivation behind most cyber attacks was similar to graffiti, in that the main purpose was to make a mark on somebody else's territory, to demonstrate technical skill by compromising a web server and defacing the main page, with the primary goal seeming to be simply to make a statement of existence. In recent years, this has evolved to being more concerned about making a profit or creating a political impact. Once the domain of the lone-wolf "hacker", cyber attacks today are more often being planned and executed by teams that have connections to criminal organizations and have profit as their primary objective.¹ From mobilizing vast bot-nets which forward spam or launch denial of service attacks, to penetrating corporate networks for embezzlement or extortion, to raiding data banks for identity information to sell, the face of cyber attacks has changed.²

Other similar scenarios have been written about, but in most cases they involved an individual attacker. This paper will attempt to describe such an event as it is orchestrated on an organizational scale. An attack by a professional organization can be expected to be quite different from the single attacker scenarios most often considered by defenders.

In October of 2001, Pat McGregor, Chief Information Security Architect of Intel, delivered a presentation entitled, "Cyberterrorism: The Bloodless War?"³ This presentation included a

slide that asserted: "InfoWarriors are not Scrip Kiddies" and showed the following bullets:

- "Funded by foreign military organizations and terrorist groups
- Likely to have more people and deeper pockets
- Can devote more resources - people and time
- They can crack systems that might withstand casual assault
- Likely to be more experienced
- Will use more sophisticated tactics
- Serious IW attackers would not reveal their activities until it is absolutely necessary"

A national security cyber attack team will have very deep resources behind it, a professional training level in the attackers' skill sets, and well thought out planning and tactics.

This paper describes a theoretical cyber attack and defense scenario between fictional organizations, using real techniques and tools for both attack and defense. The defenders will be framed as a typical network administration team responsible for the security of a large enterprise network, and using modern security standards. The fictional defending organization is a public hospital network presented as a federal government agency, known as the "Public Hospital Administration" or PHA for short. The PHA oversees the operation of public hospitals in most major cities in the U.S. They use modern information technology practices, including a nationwide network that ties all the hospitals together. Their systems are

predominantly Microsoft Windows based. They have national gateways with massive firewalls, proxy servers, enterprise anti-virus software and some level of network intrusion detection capability. The PHA uses NIST SP 800-53 as the backbone of their computer security policy.

The fictional attacking organization was actually a composite of several groups that decided to co-operate with each other for the short term purposes of this attack scenario. The driving group was an international organization of fundamental religious terrorists who wanted to strike the U.S. in any way that will create terror, make headlines and disrupt the U.S. and its economy. They planned an attack against several major cities using biological weapons. In order to maximize the effect of this attack, they decided to also launch a parallel attack against the computer infrastructure of the hospitals in the same cities. They recruited help from a secret Chinese Academy that teaches cyber attack methodology and produces over a hundred new graduates each year that are in essence, well trained professional hackers. They also recruited help from an organized crime group in Russia. The operation was financed by selling identity information harvested from the hospital network before the final attack.⁴ The Russian crime group handled this part of the operation and also any needed extortion or "muscle" operations. The Chinese group provided the cyber attackers and oversaw the entire attack operation against the computer network. In exchange, they were grandly rewarded with practical experience in the field for a select team of their graduates as well as information they highly valued on how to attack a U.S. Federal Government Agency.

The terrorist group handled the biological weapons attack and coordinated the timing of the overall operation.

The scenarios and organizations presented here are purely fictional and hypothetical, although based on real news stories and extensive documentation. The technical aspects of the tools and techniques used by both attack and defense are accurate and realistic and will be supported by evidence and references. They have either been used in a laboratory environment by the author or referenced to other sources and documentation.

2 - Preparation

A - DEFENDERS: Policy and Paperwork

Prelude to NIST (National Institute of Standards and Technology) Special Publications: In 2002, the Federal Information Security Management Act of 2002 (Public Law 107-347), gave NIST a mandate to issue Federal Information Processing Standards Publications (FIPS PUBS), which become Federal Standards once approved by the Secretary of Commerce. In March of 2006, FIPS 200 was released, which requires Federal Agencies to meet minimum information system security standards as specified in NIST Special Publication 800-53. NIST SP 800-53 also references many other SP documents that are also standards for Federal Agencies.

1. NIST Special Publications and Security Policy

The National Institute of Standards and Technology (NIST) has published a comprehensive set of documents that outline a framework of security policy and how to implement them. The heart and core of this is a "Special Publication" (SP) called NIST SP 800-53 "Recommended Security Controls for Federal Information Systems".⁵ SP 800-53 lays out 171 controls divided into 17 groups known as families. Each control consists of a definition of the scope of the control and activities that are related to the control. They usually leave open the specifics of how to implement the security control, so that different organizations can fill in different details according to their needs. A corollary document, NIST SP 800-53A⁶ contains more specific information on how to test these controls.

Some key SP 800-53 security controls, for the purpose of this paper, are as follows:

AC family - Access Control

AC-6 LEAST PRIVILEGE - in order to protect against abuse of privilege, the lowest possible level of privileges needed to accomplish tasks should be assigned to users.

AC-11/12 SESSION LOCK/SESSION TERMINATION - these two controls are designed to prevent unauthorized access to a system by initiating a time-out that locks the system and requires a user to re-authenticate. After an additional time period, it will terminate the session.

AC-18 WIRELESS ACCESS RESTRICTIONS - this control cross references SP 800-48 which goes into wireless security considerations in depth.

CM family - Configuration Management

CM-6 CONFIGURATION SETTINGS - this control suggests that settings should be configured in a restrictive mode and managed by automated mechanisms. It cross references SP 800-70. More specific configuration settings can also be found in DISA STIGS, and NIST/NSA hardening guidelines.

IA family - Identification and Authentication

IA-2 USER IDENTIFICATION AND AUTHENTICATION - this control addresses the entire area of user authentication, but for the purposes of this paper, the most interesting component is password complexity and strength, and primarily windows passwords.

IA-3 DEVICE IDENTIFICATION AND AUTHENTICATION - this control might also be considered "port level security", as it talks about authenticating devices on the network. In plain words, if you can plug any device into any network port and get connectivity without any type of authentication, this control is not being used.

PE family - Physical and Environmental Protection

PE-3 PHYSICAL ACCESS CONTROL - deals with how physical access is controlled including "publicly accessible" areas.

RA family - Risk Assessment

RA-3 RISK ASSESSEMENT - offers a very generalized framework and cross-references NIST SP 800-30 for details. SP 800-100 also offers a more detailed scheme for handling risk assessment.

PL family - Security Planning

PL-2 SYSTEM SECURITY PLAN - calls for a security plan that outlines the system involved and the security controls needed to protect the system and cross references SP 800-18.

SI family - System and Information Integrity

SI-2 FLAW REMEDIATION - this control specifies the need for automated and centrally managed patch and update management in a general sense. In actual implementation, the most important component of this control may be how the organization handles Microsoft security updates.

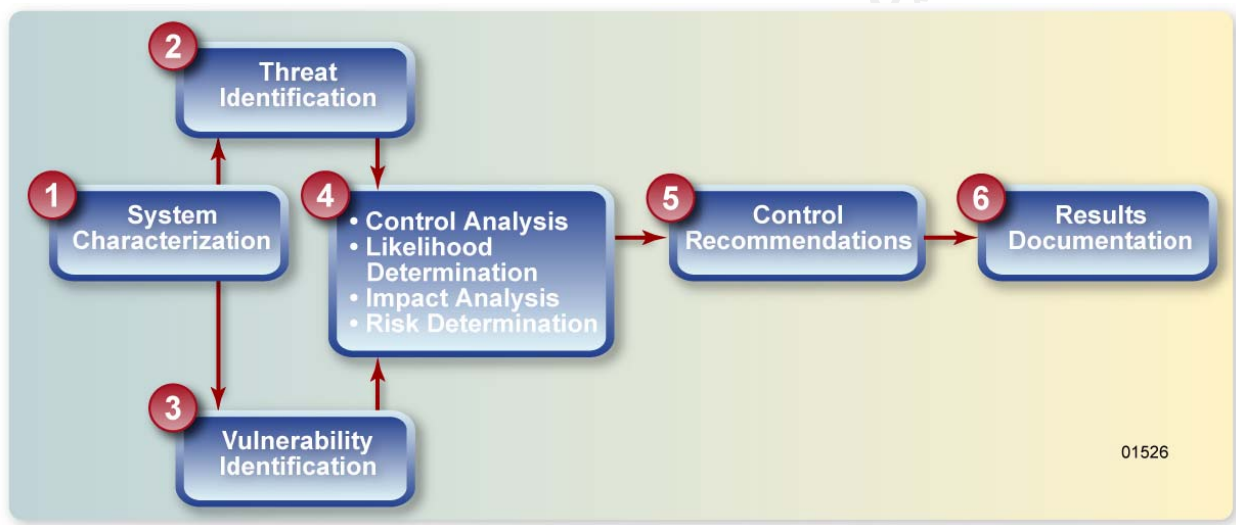
SI-3 MALICIOUS CODE PROTECTION - this control describes the needs for automated and centrally managed anti-virus protection mechanisms that include automatic updates.

SI-4 INTRUSION DETECTION TOOLS AND TECHNIQUES - this control requires that the organization performs Intrusion Detection and offers some guidance but leaves most of the details open. SP 800-94 is "Guide to Intrusion Detection and Prevention Systems".

The CA family, "Certification, Accreditation and Security Assessments" specifies how a C&A process should be accomplished. This includes; assessing policies and procedures, testing controls, remediation plans, continuous monitoring and more. The assessment control cross references 800-53A. Federal Agencies are required to complete a C&A process every three years and maintain monitoring and updates in the intervening years. Controls from this family will not be discussed in this paper, but it's important to note that the C&A process is quite intensive, is required by law for Federal Agencies and often diverts much attention and effort away from actually strengthening network defense, instead concentrating it on completing paperwork to get certification. Without the appropriate certification, the IT infrastructure of an agency does not have authority to operate. Greater detail for this entire family is also found in NIST SP 800-37 "Guide for the Security Certification and Accreditation of Federal Information Systems".

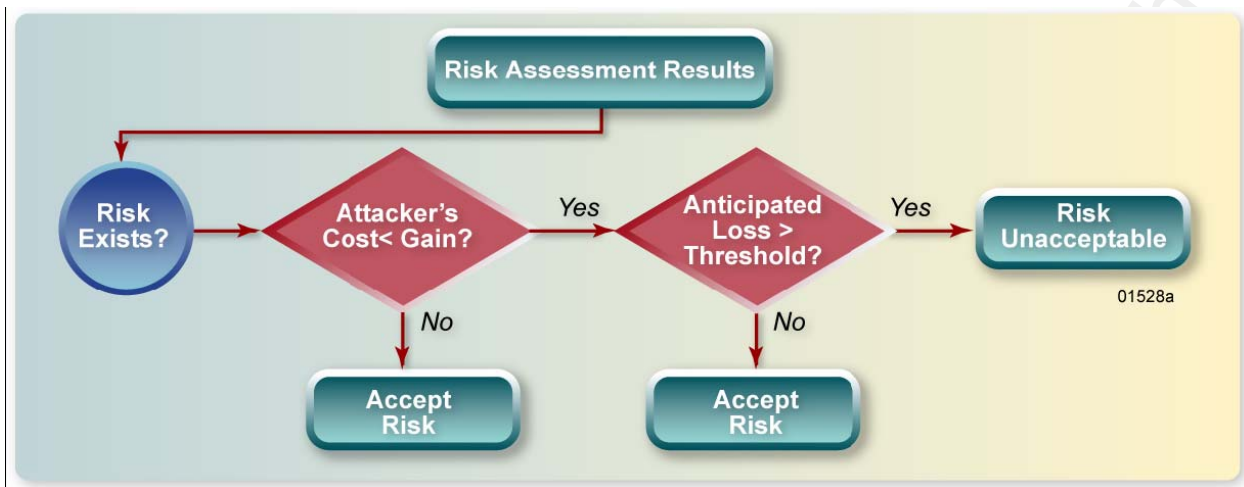
Another NIST SP document, 800-100 "Information Security Handbook: A Guide for Managers"⁷ gives us a good look at the overall process of putting together a security plan and controls to secure a system. Chapter ten, "Risk Management" explains a process of identifying threats and vulnerabilities then using controls to mitigate risk. The likelihood of success of a particular attack

against a particular vulnerability needs to be weighed against any possible impact to come up with an overall determination of risk level. Control recommendations can then be developed that tailor the security plan to respond to this risk assessment.



[diagram from NIST SP 800-100 for "Risk Assessment Process"]

This process is designed to consider both threats and vulnerabilities and weigh them together, producing an assignment of a risk value.



[diagram from NIST SP 800-100 for "Risk Mitigation Strategy"]

This process helps to decide whether or not a risk can be accepted, thereby enabling decisions on the approach to mitigation strategies and focus on controls.

2. RA-3 RISK ASSESSMENT (defender's policy)

[the italicized section below is a security control from NIST SP 800-53]

Control: The organization conducts assessments of the risk and magnitude of harm that could result from the unauthorized access, use, disclosure, disruption, modification, or destruction of information and information systems that support the operations and assets of the agency.

Guidance: Risk assessments take into account vulnerabilities, threat sources, and security controls planned or in place to determine the resulting level of residual risk posed to organizational operations, organizational assets, or individuals based on the operation of the information system. NIST Special Publication 800-30 provides

guidance on conducting risk assessments including threat, vulnerability, and impact assessments.

[the following section is the PHA response to the security control described above]

Implementation: A risk assessment of each facility was performed in 2006 and will be updated again in 2007. The risk assessment was designed to identify the threats and vulnerabilities of the system. It was performed using an automated tool that inputs the answers from risk assessment questions, calculates the risk, and produces reports. The risk assessment report for each facility is to be kept in a locked container and marked, "Sensitive Data". The data in this report is used to support the Certification and Accreditation determination of risk.

3. PL-2 SYSTEM SECURITY PLAN

[the italicized section below is a security control from NIST SP 800-53]

Control: The organization develops and implements a security plan for the information system that provides an overview of the security requirements for the system and a description of the security controls in place or planned for meeting those requirements. Designated officials within the organization review and approve the plan.

Guidance: NIST Special Publication 800-18 provides guidance on security planning.

[the following section is the PHA response to the security control described above]

Implementation: The format for the System Security Plan (SSP) was developed by the PHA Certification and Accreditation project under the Office of Information Architecture and authorized by the Deputy Assistant Secretary for the Office of Information. The plan has been reviewed by the Office of Cyber Information and Security Compliance Assurance. Ensuring that the plan is kept up to date and current is the responsibility of the owner of each system.

4. Threat Analysis

The following is an excerpt from a document produced by the PHA, analyzing the threats that should be considered in a hospital network.

Threat Profile of a Network of Hospitals

Executive Summary:

In the normal day to day operation of a network that supports a group of hospitals, the most critical asset to the mission of the hospitals is patient information. If the patient information is incorrectly modified or missing, the potential exists for loss of human life. In a modern cyber attack against a network that supports a group of hospitals, the most valuable asset to the attacker is patient information. The attacker might have a motive to disrupt hospital operations by interfering with the availability of the patient information, or the motive might be simply financial gain from selling identity records harvested from the network. It is also possible for an attacker to embrace both motives simultaneously.

Analysis of the threat profile produces the following as major areas of concern:

- Accidental disclosure/modification/destruction of patient information by insiders, outsiders, malicious code, or infrastructure failures.
- Intentional disclosure/modification/destruction of patient information by insiders or outsiders.

Accidental issues are already largely mitigated by system controls, data backups, redundant power supplies and other conventional defenses against natural disasters. Accidental damage is also far more likely to be limited to a local area.

This leaves intentional issues as the major threat vector. Loss analysis predicts possible large scale loss of life caused by nationwide disruption of the network and/or the possibility of many millions or even billions of dollars in financial gain for the attacker by selling identity records.

Vulnerability Analysis:

Physical security is weak at most hospitals since most areas of a modern hospital are open to access by the public. Even when certain areas of hospital space are "off-limits" to the public, both security measures and staff awareness of security considerations is very low. In most cases, an intruder can freely explore all areas (except where sterility is required) without being challenged.

Physicians' workrooms (common office areas with shared computer and printer access) are of particular concern, since the doctors usually walk away from a computer system when they are finished using it without logging off from the system.

Network security is also weak. Most hospitals do not patch security holes announced to the public within forty-eight hours, do not have strong enough authentication procedures (password complexity and storage is an issue), or access control (session timeouts and port control are issues). Configuration hardening is not done well, and Intrusion Detection is either absent or lightly monitored. More and more hospital equipment is using wireless connectivity with all the vulnerabilities attached to it. Enterprise Anti-virus installations may be the only network security strong point found in a modern hospital network, but as targeted attacks using customized malware (malicious software such as viruses, worms or trojans) become more common, its effectiveness is dropping quickly. Data encryption is becoming more common on laptops, but is rarely found anywhere else on the network.

5. Scenario: Defenders' Policy

The process of identifying threats and vulnerabilities and weighing them is critical to success in any later defense efforts. If the threat analysis is not done correctly, the security plan and the controls that are selected might not be appropriate. Most large organizations find it easy to get this right when considering natural disasters, because they have been dealing with them for many years and often build up local experience and expertise. A hospital

located along a hurricane prone coastline understands that during a serious storm, their utility grid power supply will be out and the bottom floor of their facility will be flooded, so the backup generator and any IT infrastructure need to be located on floors above some flood line. Likewise, facilities located in northern regions deal with snowstorms and cold conditions gracefully and so on.

Many security plans today almost ignore cyber attack threats and almost all of them fail to take the threat seriously enough. SP 800-100 says, "In other words, it is not possible to estimate the level of risk posed by the successful exploitation of a given vulnerability without considering the efficacy of the security controls that have been or are to be implemented to mitigate or eliminate the potential for such an exploitation; **nor the threat's motivation, opportunity, and capabilities**, which contribute to the likelihood of a successful attack; **nor the impact to the system and organization should successful exploitation of a vulnerability occur.**"⁸ (bold added for emphasis)

An analysis of security controls (based on SP 800-53A) is supposed to be done simultaneously with the rest of the risk management process in order to help determine the likelihood of success of a particular threat. This is almost never done and this case was no exception. The site security plan was assembled by cutting and pasting from a template distributed by the organization and it was composed by members of the security team with more experience in handling the legacy problems (such as natural

disasters) and little awareness of modern cyber attack technology and methodology. As a result, special focus was given to controls that mitigate the threats perceived as being most important and other controls related to cyber attacks were given little attention, instead simply producing the paperwork needed for certification.

The threat analysis paper excerpted above seems to have identified some serious threats to the hospital network, but the focus and security controls that should be expected as a result were not noted in any other documentation. The threat analysis seems to have gotten lost in the bureaucracy of a large government agency and not used. Unfortunately, it was published on a public web page.

The defenders' policy was encyclopedic in its size and volume, and only a few small excerpts are represented here. It was also quite sketchy and vague when it came to specifying the details needed in order to actually defend the network. It almost seemed as though many of the policies were written because there was a requirement to have one, and so the wording was selected to meet expectations instead of being aimed at enforcing security controls designed to contain vulnerabilities.

B - ATTACKERS:

1. Col. Boyd's OODA Loops⁹

"Speed is the essence of war. Take advantage of the enemy's unpreparedness; travel by unexpected routes and strike him where he has taken no precautions."¹⁰ Sun Tzu

An Air Force Colonel named John Boyd achieved a reputation as a talented fighter pilot and then went on to become one of the best pilot instructors at the Fighter Weapons School at Nellis Air Force Base. Col. Boyd earned the nickname, "forty second Boyd" by making a standing offer of \$40 for anybody who could survive for forty seconds against him in an aerial dogfight, starting out from a position on Boyd's tail. He never lost that bet. Boyd went on to become a key figure in the design of both the F-15 Eagle and F-16 Fighting Falcon fighter planes and after he retired, gave briefings and lectured about combat maneuvers to many military groups and training organizations.

What Boyd became most known for was developing a theory of the timing involved in combat maneuvers. He called it "OODA" for; Observe, Orient, Decide, Act. Boyd's theory says that every combat maneuver has to constantly loop through these actions and whoever can perform the cycle fastest gains a distinct advantage. It's easy to understand how valuable this theory is when it pertains to aerial combat. Pilots engaged in a dogfight must be able to very quickly transition from seeing an action to making some kind of sense of the action, to making a combat decision, to taking the action necessary

to evade an opponent, to reverse positions, or to kill their opponent. But this theory can also be applied to almost any other form of combat, whether on land or sea, whether it deals with single opponents or large groups. In this case, it can also be applied to information warfare.

Although speed was clearly at the core of OODA loop theory, Boyd went beyond that to also focus on variety, harmony and initiative as opponents continuously cycled through their combat loops. Variety in your techniques makes it difficult for your opponent to orient and slower to decide. Harmony across your techniques increases your speed of response. Initiative can put you in the aggressive lead of the loop and force your opponent to a defensive position where a mistake can become fatal.

While most military training focuses on a two dimensional landscape, Boyd's experience as a fighter pilot required that he think in three dimensions. It is logical to extend this to include time. In other words, you must learn to maneuver in time as well as in space. In fighter pilot terms, this means that instead of following the ever curving path of an enemy fighter trying to evade your guns, you have to get "inside" his loop, or anticipate where he will be at a future point and take a shorter path to arrive there in time to destroy him. Boyd's theory was that all combat maneuvers must be designed to "get inside" the opponents OODA loop, whether in space, time, information, psychology, or combinations of these factors. Boyd once said, "Machines don't fight wars. Terrain doesn't

fight wars. Humans fight wars. You must get into the minds of humans. That's where the battles are won."¹¹

In order to create faster speed (as well as increased variety, harmony and initiative), it is necessary to train combatants to high levels of proficiency in each of the phases of the OODA loop. Observation and Orientation come first and enable the Decision and Action phases. Being able to correctly frame and understand what you are "seeing" is also known as Situational Awareness (SA). Having good SA opens up opportunities to maintain harmony within your own actions and introduce variety and initiative that can confuse your opponent and actually slow down their ability to cycle through their own OODA loop process.

2. Situational Awareness Matrix

"So it is said that if you know your enemies and know yourself, you will win hundred times in hundred battles. If you only know yourself, but not your opponent, you win one and lose the next. If you do not know yourself or your enemy, you will always lose." Sun Tzu

In any combative scenario, situational awareness is a key to the outcome. It's not just knowing where you are and where your opponent is, but also what condition and state each of you are in and details about the environment and obstacles you both face.

Applied to network cyber attacks, a situational awareness matrix can be developed and filled in as the attack progresses. A simplified version of the matrix might look like this:

| | EXTERNAL tactics | PHYSICAL tactics | NET border tactics | NET interior tactics | HOST tactics | PEOPLE tactics |
|----------|---------------------|---------------------|-----------------------|-------------------------|-----------------|-------------------|
| Recon | info-gather | observe | scan | sniff | direct | info-gather |
| | | | vscan | fingerprint | | observe |
| | | | sniff | direct | | trash |
| Intrude | | break-in | console | console | direct | bribe |
| | | walk-in | protocol | protocol | exp code | extort |
| | | | crack | crack | crack | torture |
| | | | | | bypass | |
| Entrench | | | de-log | de-log | de-log | |
| | | | access | access | access | |

The attack begins with external reconnaissance and progresses inward, to intruding across the network with the intention of eventually compromising interior host systems and then beginning a "PIVOT" attack to use the compromised host as a base to attack other systems inside the network. At the same time as the attack penetrates from the exterior to the interior, it also progresses in tactics from reconnaissance to intruding and once they have gained a foothold on a compromised system, they take action to entrench this position and make sure they can regain access to the system at a later time.

Each individual attack would not visit all parts of this matrix, but would follow its' own unique pathway through the matrix. We can further expand the matrix by adding in tools to be used with each tactic.

| | NETWORK | | Interior | | HOST | |
|----------|----------|------------|--------------|-------------|---------|------------|
| | tactics | tools | tactics | tools | tactics | tools |
| Recon | identify | whois | sniff | wiresh/yers | | |
| | identify | nslookup | scan | nmap | | |
| | scan | nmap | sniff | ettercap | | |
| | vscan | nessus | vscan | nessus | | |
| | sniff | kismet | p-fingerprnt | p0f | | |
| | webscan | nikto | p-fingerprnt | xprobe | | |
| | frag | frag-route | direct | show info | | |
| | scan | firewalk | | | | |
| Exploit | | | | | direct | manual |
| | | | | | exploit | single |
| | | | | | exploit | metasploit |
| | | | | | crack | cain |
| | | | | | crack | john |
| | | | | | bypass | cust-malw |
| Entrench | | | | | de-log | manual |
| | | | | | access | backdoor |
| | | | | | access | metasploit |
| | | | | | stealth | rootkit |
| | | | | | stealth | stego |
| | | | | | stealth | covert ch |

[for instance: the scan tactic might use nmap as a tool, the vulnerability scan tactic might use Nessus as a tool, the sniff tactic might use Wireshark, while another sniff tactic focused against a wireless target might use kismet instead]

We can also add defensive tactics and tools that are anticipated and the appropriate counter measures. An attack "war-board" based on the situation matrix could be set up in a command and control facility to track the progress of an attack and collect information regarding the target and defenses as it is learned. In a team situation, being able to quickly relay such information from one unit to another in a live attack is critical.

Members of the attacking units need to be trained to constantly think about the situational awareness matrix and ask themselves the questions:

- Where are you? In a cyber sense, where on the network are you, what system are you on, what kind of system is it?
- What do you know? What can you "see", what network protocols are being used, what services are running, what ports are open?
- What can you access? Where can you reach from where you are, what limitations are there on what you can access?
- What can you control? What can you take control of, what can you not take control of?
- What do the defenders know about you? Do they have any information that might indicate that you are on their network, what tools are at their disposal that might disclose your activity?
- How will the defenders react if they discover your activity? What do you expect the scaling of their reactions to be and what actions will they take at each level?

The state of situational awareness (SA) will have a great impact on the ability of the team to quickly cycle through Boyd's OODA loops. This speed advantage creates a competitive edge that allows the attackers to elude detection or eradication and create confusion among the defenders. Tailoring the attack tactics and strategy to facilitate SA and OODA loop theory might mean making email servers a priority target for the purpose of intercepting email that describes

defensive activity. Incident response methodology stresses using "out of band" communications mediums for exactly this reason.

3. SCENARIO - "The Academy"

The Chinese Hacking Academy was designed to teach advanced cyber attack and penetration techniques to qualified students. Students were selected based on a mix of fundamental computing and networking skills, an interest in penetration techniques and a unique psychological profile that included a creative element and a viewpoint that saw obstacles as a challenge. This can often manifest itself as an anti-authoritarian attitude and may result in the student being labeled as a trouble-maker. These types of students were sought out and examined carefully for suitability to the program.

The "Master" of the academy was well versed in Sun Tzu, the theory of maneuver warfare by Clausewitz, Boyd's OODA loop theory, and the U.S. Marine Corp manual called, "FMFM1 Warfighting". He taught the philosophy of cyber attacks as much as tactics and technique and oversaw the other instructors in the school.

Training Syllabus:

Basics

Network protocols refresher

Beginning packet analysis

Basics of intrusion detection signatures

Anti virus signatures

Intro to reverse engineering

Fundamental reconnaissance

How email servers work

IIS versus Apache - web server basics

Incident handling methodology

Advanced

Datagram fields

Advanced IDS analysis

Port scanning and assessing vulnerabilities

Passive fingerprinting

Wireless security assessment

Buffer overflows and format strings

Penetration tools (metasploit, canvas and core impact)

Backdoors

Web and SQL attacks

Botnet command and control and DDOS attacks

Forensics

Encryption

VPN technology

Sessions: Man In The Middle attacks and hijacking

Counter defensive

Password cracking

Using fragmentation to elude IDS

Counter forensic measures

Root kits: from user mode to kernel mode

Covert channel communication

Stealth using polymorphic techniques

Steganography techniques

In addition to the lectures on cyber attack philosophy and situational awareness, the students were trained in attack techniques and all the commonly available tools plus some custom made versions. They were well schooled on both the attack and defense sides of each phase of cyber conflict. For instance, a student would be taught how to use an exploit to compromise a system, then an intrusion detection system would be introduced that could notice the compromise, then the exploit code was obscured using polymorphic techniques, then IDS techniques were used that can detect payload anomalies statistically. Then the payload was further obscured and the entire exploit was packaged in a wrapper designed to deliberately trigger an old exploit signature such as Code Red from 2001. The assumption was that the defenders were not capable of reacting in real time, giving the attackers time to "dig in" with a stealthy root kit. The defenders would not be likely to investigate very thoroughly, thinking that since Code Red was such an old exploit, and the system was patched against that many years ago, there was no real danger. At each stage

in the training, cyber gaming exercises were used to sharpen skills and evaluate progress.

An excerpt from a U.S. Marine Corps manual of military philosophy, "FMFM1 Warfighting" best describes the teaching philosophy of the Academy:

"Maneuver warfare is a warfighting philosophy that seeks to shatter the enemy's cohesion through a series of rapid, violent, and unexpected actions which create a turbulent and rapidly deteriorating situation with which he cannot cope.

From this definition we see that the aim in maneuver warfare is to render the enemy incapable of resisting by shattering his moral and physical cohesion--his ability to fight as an effective, coordinated whole--rather than to destroy him physically through incremental attrition, which is generally more costly and time-consuming. Ideally, the components of his physical strength that remain are irrelevant because we have paralyzed his ability to use them effectively. Even if an outmaneuvered enemy continues to fight as individuals or small units, we can destroy the remnants with relative ease because we have eliminated his ability to fight effectively as a force."¹²

3 - Using Google for Reconnaissance

A. Scenario (Google)

While conventional information reconnaissance used to begin in a public library, it now begins with public online materials. In this case, many large public organizations have either home web sites and/or wiki articles that include many different forms of information. Hospitals and government agencies are no different. The attackers started out with general searches to build a list of main web sites related to their targets, and also any news stories or wiki pages that could be found.

Using these basic techniques, it was possible to retrieve the following information:

- Maps of facility locations and directions to them.
- Maps of facility buildings and satellite images.
- Maps of building interiors showing departments and function of areas.
- Organizational charts of departments and staff positions.
- Lists of staff including contact information.
- Job listings with descriptions of technical skills needed.
- Help desk Frequently Asked Questions.
- News stories.
- Security policies.

The "Enterprise Information Security" page was especially helpful, yielding the name of a file integrity checker program and the name of a popular anti-virus vendor. Policies were posted that offered a list of ports and protocols that were allowed on both the wired network and the wireless network. The wireless networking policy explained that the facility was using both 802.11b and 802.11g and WEP encryption. In some cases the links were protected by an account login process, but just the name of the link provided the information needed. Two of the URLs in links that could not be reached without logging in, showed that the facility was using Tripwire software for file integrity checking and Microsoft's SUS (System Update Server) to distribute patches and updates. A "Troubleshooting" section patiently explained that the password required to un-install the anti-virus software was the name of the manufacturer (note - this is a known default setting).

A "Network Configuration" page offered IP addresses for routers used as default gateways, DNS servers, SMTP servers, and a link to a separate page that had a list of printers, their locations, make and model, serial number, purchase data and amount of RAM contained in the printer. Another page linked to this one showed a list of maps that went floor by floor in some buildings and marked the location of every ethernet jack. There was also a list of ethernet jacks available in conference rooms for public use. At one facility, a list was discovered on a web page named "department servers - essential machines". It contained IP addresses, host names, operating systems and version, make and model and serial number of hardware, primary function, physical room location, CPU speed and RAM

amount and hard disk capacity.

By noticing that the keywords seemed to be "department servers", the attackers drilled in some more google searches and found a lot more. Some more FAQ pages turned up and by exploring the links there, they produced more policy and procedure regarding computer use, network diagrams, and a section labeled "photos of A-Wing communication closets - category three punch-down block information". Clicking on the link to view the photos returned a 403 Access Forbidden message, but the links contained the room numbers of all the wiring closets in the building. There was a list of network admin staff, their office locations and phone numbers. A "What's New?" link offered a change-log of upgrades to application software and network services, including the version and date of the change. A "System Admin Utilities Software" page offered even more details on the current versions of many tools being used. Spinning off another search branch using the key words "network diagram" was also very fruitful.

A major windfall discovered on one of the web sites was the PHA 6601.1 Information Security Handbook in a .doc file. This document was over fifty pages long and included a lot of material that appeared to have been copied and pasted directly from NIST SP 800-53 and dealt mainly with policy. Several long appendices were also obtained and one of them contained security controls and configuration information. It referenced all of the 800-53 controls and although many of the descriptions of implementation of the controls were vague and generalized, some of them were quite

explicit. The exact configuration of the password complexity policy was available. The controls that were vaguely worded could mean either that the policy makers or the front line defenders didn't understand them very well, and might offer some opportunity for the attackers.

Another document retrieved from the same site seemed to define the PHA policy for Incident Response (IR), but it had not been updated since 1999. It did offer a good framework for general IR procedures and vaguely defined conditions under which the Information System Security Officer and the Facility Director were to be notified. Most of the focus on cyber threats seemed to be directed toward contamination by viruses. The PHA 6601.1 handbook also referenced this document for IR controls.

News stories about new hires provided tremendous biographical background information about key staff members. A local newsletter article showed a picture of a computer technology class engaged in a computer security training exercise. At first glance this seems innocuous, but it yielded details like the name of the professor of the class, the name of the campus building the class was held in, and several names of students in the class. Another news story told of the facility's "Crisis Response Team" including some of their security measures and the name of the chairman of the team. This kind of material was collected in great volume and poured into a database for later correlation and reference for social engineering. Biographies for key staff members were developed that included home addresses and phone numbers and past jobs and contacts. Particular

intensity was applied to this search whenever a biography was identified as being a possible "key" staff member (somebody that might have important knowledge or access rights to key infrastructure). Some professional networking sites were very helpful with this. Searching with keywords, "JCAHO" and "HIPAA" yielded hundreds of contacts who were selectively invited to link to a fictional account, which was then able to request invitations to colleagues of the links.

Application produced data files (.doc, .xls, .ppt, .pdf) were downloaded and analyzed and produced metadata information on the documents' authors and editors and sometimes their contact information. A few even contained internal host names. Images from Google and Flickr also supplied many useful photographs of sites, buildings, office space interiors and people. Captions often provided the names of the places and people in the pictures. On the Facilities Management page, a fire alarm testing schedule was posted for months in advance of the testing dates. Who knows if this would be useful or not? It was added to the database.

Conventional reconnaissance was used as a follow up after the initial online searches. In some cases, the medical facilities were part of or associated with universities and the campus library became a helpful resource. Telephone queries were made using the contact information discovered above to confirm current staff status and even some technical details. Agents were sent onsite to perform physical observation and take pictures when it was needed to fill in the blanks. They also developed a list of local coffee shops, delis and

restaurants within walking distance that were determined by following facility staff on foot.

More detailed reconnaissance searched through "google groups" postings, and found computer network security policies posted on web pages. Many email addresses were harvested using combinations of the site domain name plus "@gmail.com", "@hotmail.com", "@yahoo.com" ... and so on.

Some more detailed google searches were made to find systems and vulnerabilities. In google advanced search, the results were narrowed to the domain name, then a search was done for "Apache/ server at". Here are some of the results:

- Apache/2.2.4 (Fedora) Server at domain.name Port 80
- Apache/2.0.52 (Red Hat) Server at domain.name Port 80
- Apache/2.0.52 (CentOS) Server at domain.name Port 80
- Apache/2.0.49 (Unix) mod_ssl/2.0.49 OpenSSL/0.9.7c PHP/4.3.2 Server at domain.name Port 80
- Apache/2.0.47 (Unix) mod_perl/1.99_10 Perl/v5.8.0 mod_ssl/2.0.47 OpenSSL/0.9.6g Server at domain.name Port 80
- Apache/2.0.46 (Red Hat) Server at domain.name Port 80
- Apache/1.3.37 Server at domain.name Port 80
- Apache/1.3.33 Server at domain.name Port 80
- Apache/1.3.29 Server at domain.name Port 80

- Apache/1.3.27 Server at domain.name Port 80
- Apache/1.3.26 Server at domain.name Port 80
- Apache/1.3.20 Server at domain.name Port 80
- Apache/1.3.9 Server at domain.name Port 80

There are several server versions on this list that are vulnerable to the very old "Apache Web Chunked" exploit. Note that there may have been several or even many Apache servers for each listing above. Only one listing was shown for each version detected, regardless of how many were seen.

4 - Perimeter

A. DEFENDERS: SI-2 FLAW REMEDIATION

[the italicized section below is a security control from NIST SP 800-53]

Control: The organization identifies, reports, and corrects information system flaws.

Guidance: The organization identifies information systems containing proprietary or open source software affected by recently announced software flaws (and potential vulnerabilities resulting from those flaws). Proprietary software can be found in either commercial/government off-the-shelf information technology component products or in custom-developed applications. The organization (or the software developer/vendor in the case of software developed and maintained by a vendor/contractor) promptly installs newly released security relevant patches, service packs, and hot fixes, and tests patches, service packs, and hot fixes for effectiveness and potential side effects on the organization's information systems before installation. Flaws discovered during security assessments, continuous monitoring (see security controls CA-2, CA-4, or CA-7), or incident response activities (see security control IR-4) should also be addressed expeditiously. NIST Special Publication 800-40 provides guidance on security patch installation.

Control Enhancement 1: The organization centrally manages the flaw remediation process and installs updates automatically without individual user intervention.

Control Enhancement 2: The organization employs automated mechanisms to periodically and upon command determine the state of information system components with regard to flaw remediation.

[the following section is the PHA response to the control described above]

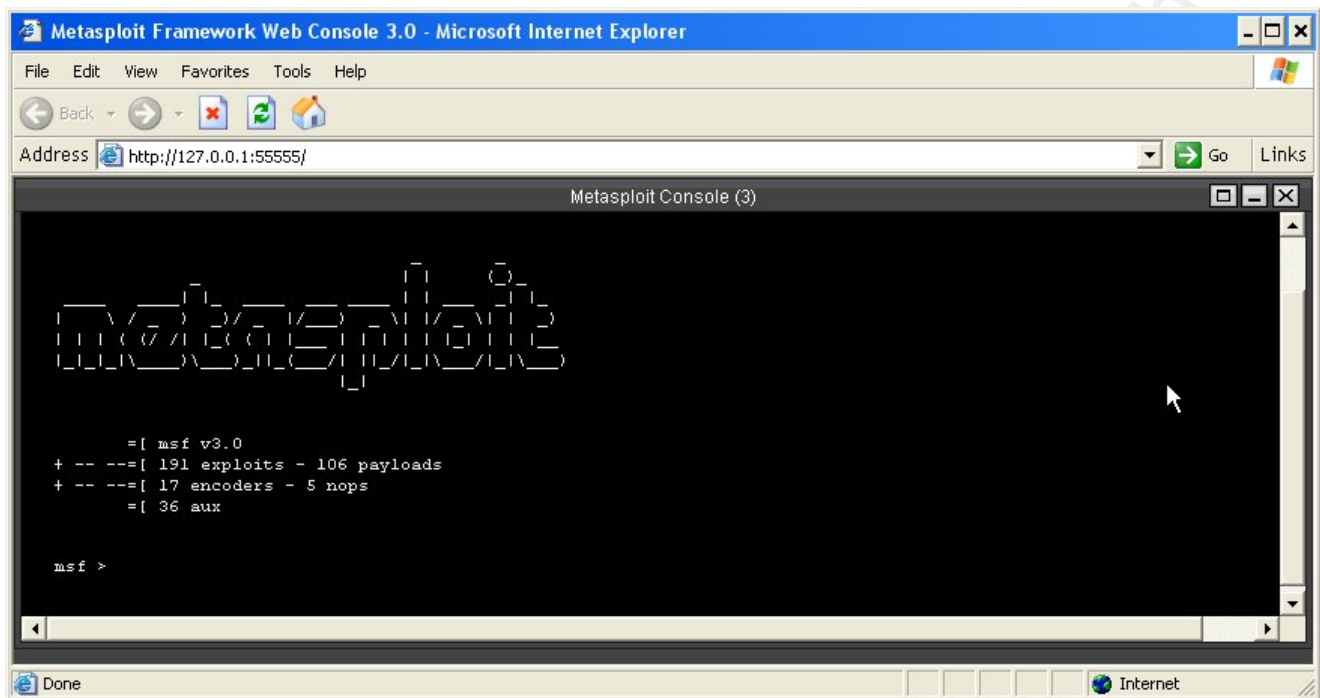
Implementation: PHA requires that operational units identify, report on, and correct information system flaws by installing updates, as appropriate.

Automated mechanisms are to be used to periodically determine the state of systems with regard to updates and to install the updates without individual user intervention. This includes security patches, service packs, and hot-fixes. They must be installed in a reasonable timeframe or in accordance with guidance as issued.

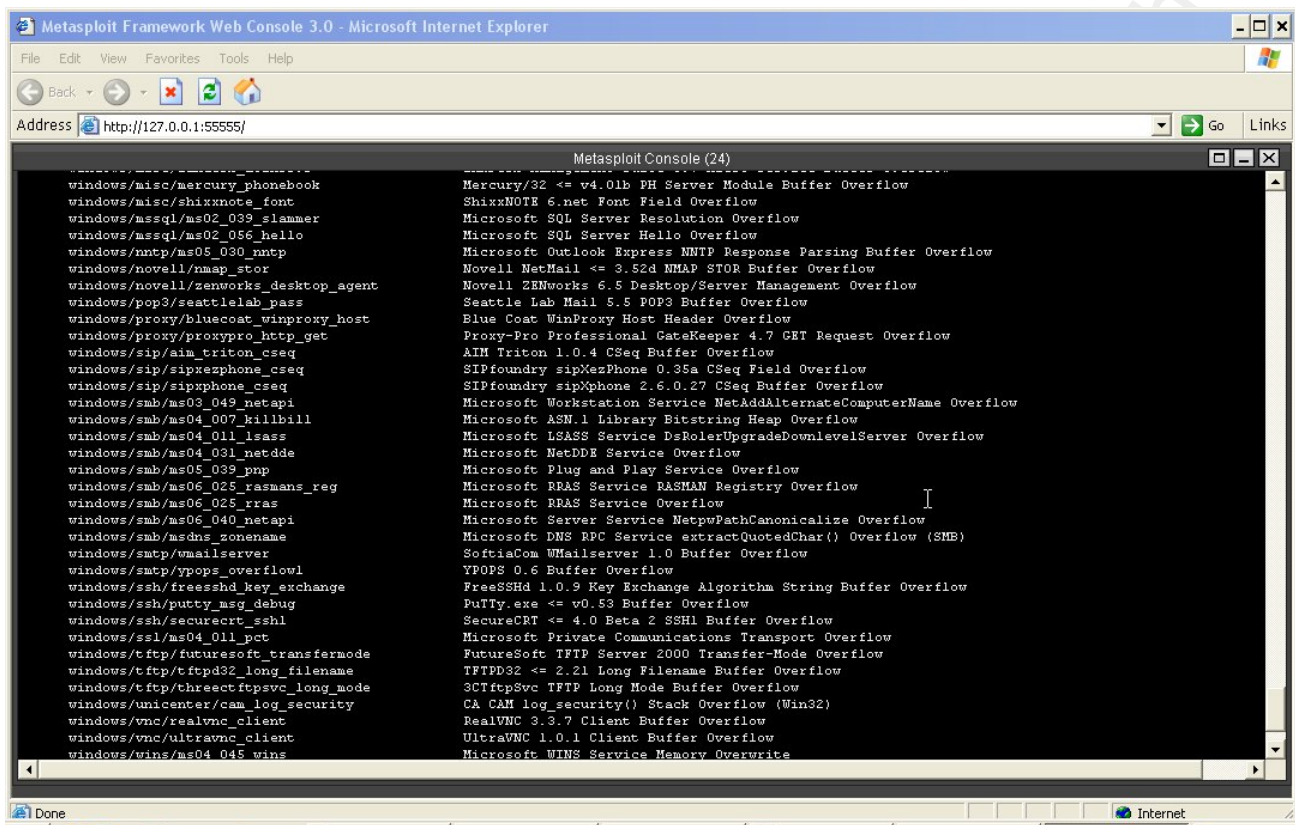
Documentation is to be maintained that shows compliance with the requirement that system updates are being identified and installed in a reasonable timeframe and expeditious manner, and that control responsibility has been assigned and specific actions taken to ensure implementation that consistently applies flaw remediation efforts and that all appropriate information is captured and recorded pertaining to the discovered flaws, including the cause, mitigation activities and lessons learned.

B. ATTACKERS: METASPLOIT

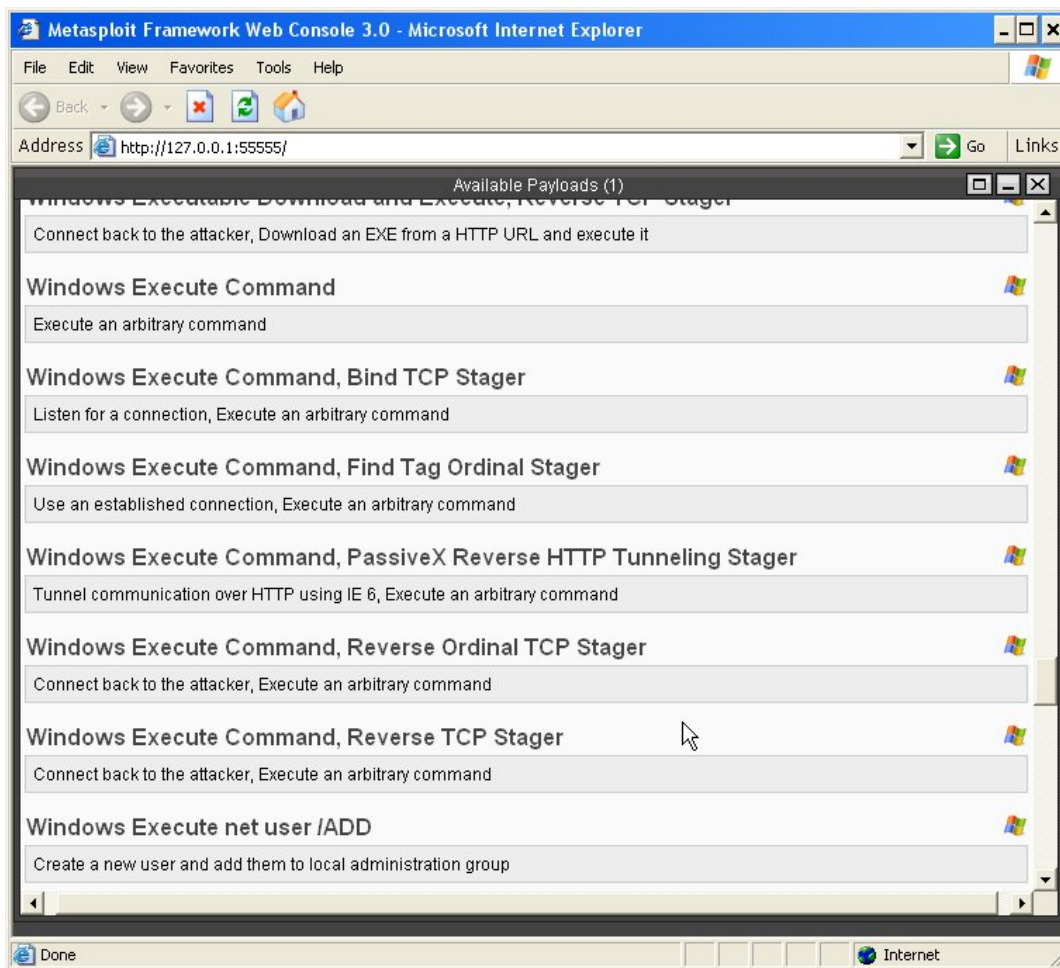
Metasploit¹³ is a framework with modular exploit and payload components used to compromise vulnerabilities and penetrate systems. Both command line and web based interfaces are available. Once a system vulnerability has been identified, exploit code that matches the vulnerability is selected from a list and some targeting parameters such as the IP address of the target are set.



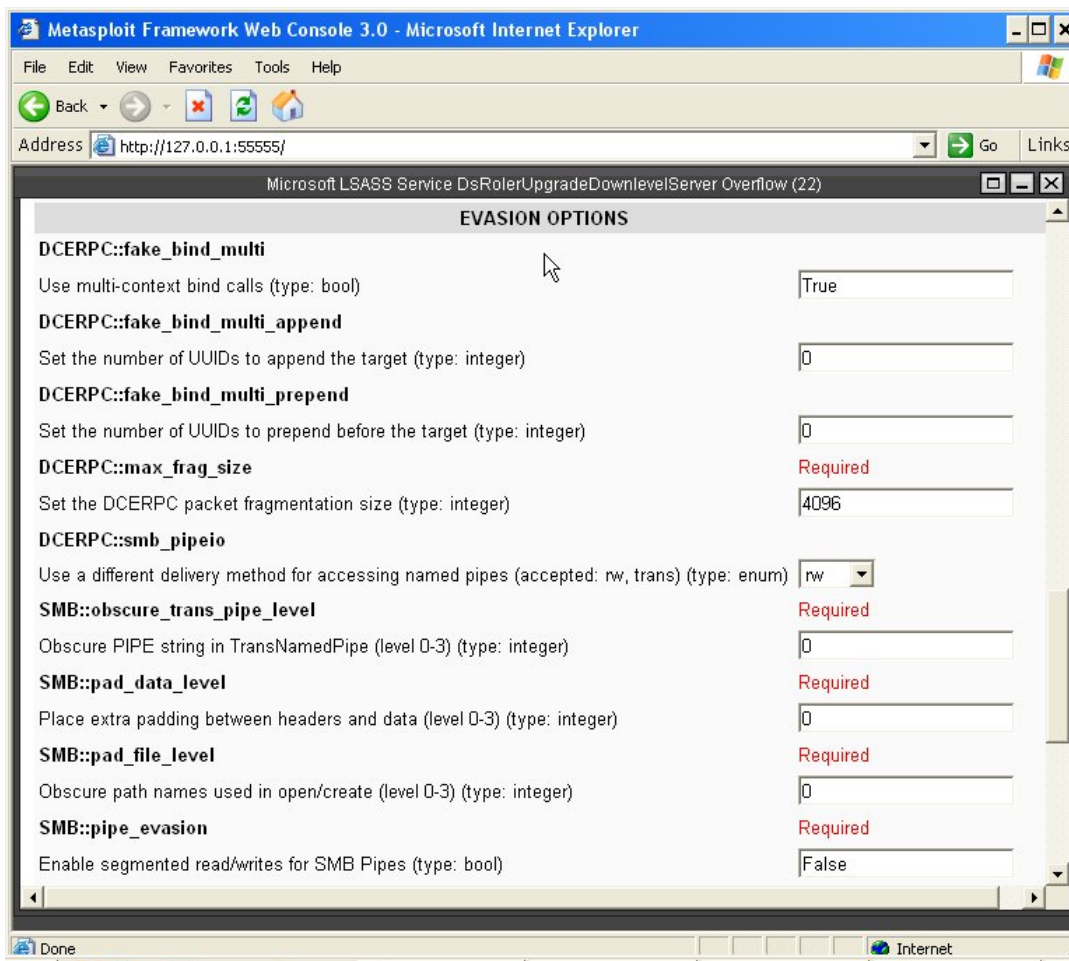
[there are 191 exploits and 106 payloads in this version - the command line interface shown here]



[some of the exploits - command line interface again]

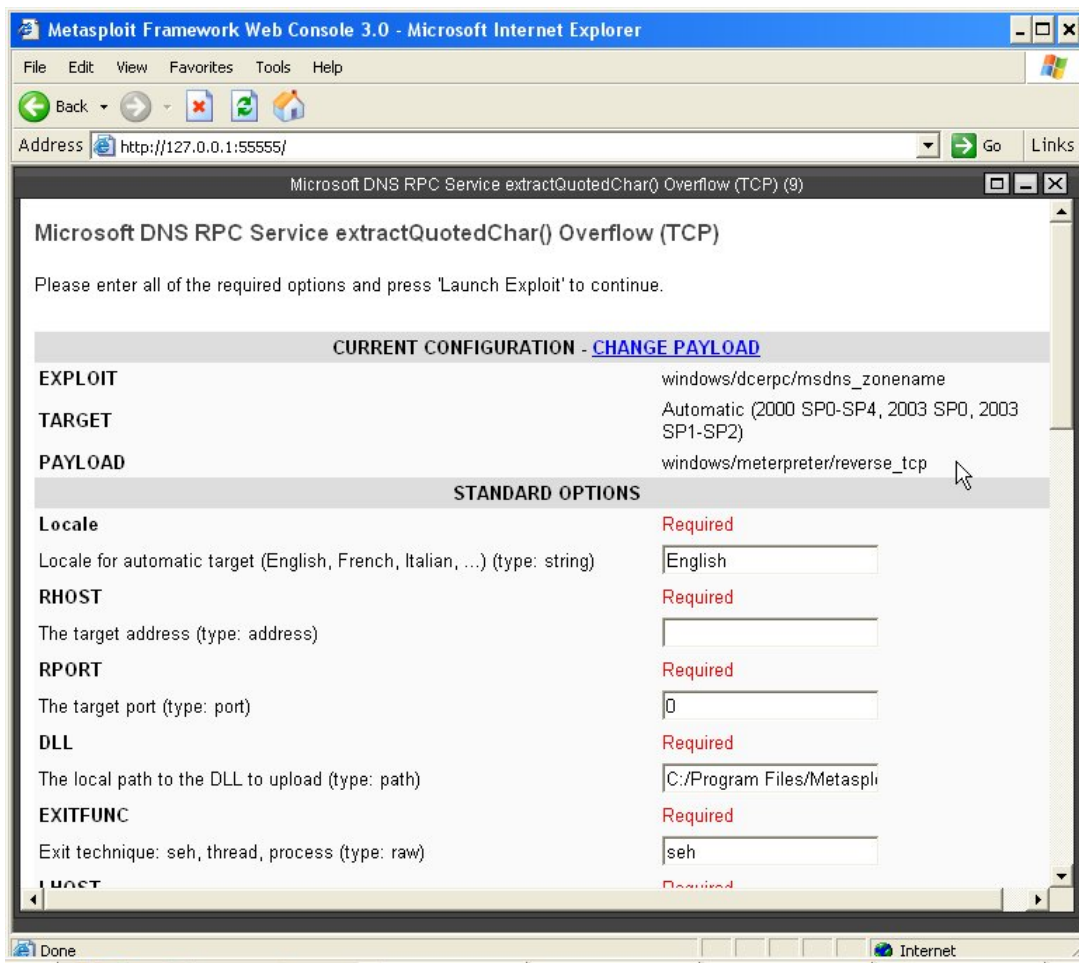


[some of the payloads – web interface shown here]



[In many cases, evasion options can be set – web interface again]

A payload is selected and configured and the exploit is launched. If the exploit successfully penetrates the system, the payload typically returns command prompt access to the attacker. Metasploit also includes a multi-function payload with advanced features called "Meterpreter". It runs entirely in system memory and acts as a command interpreter. It can use encryption to remain stealthy, can transfer files, dump password hashes and many more functions. It is extensible and can load DLLs to provide more functionality.



[payload options – web interface shown]

C. Scenario (Perimeter)

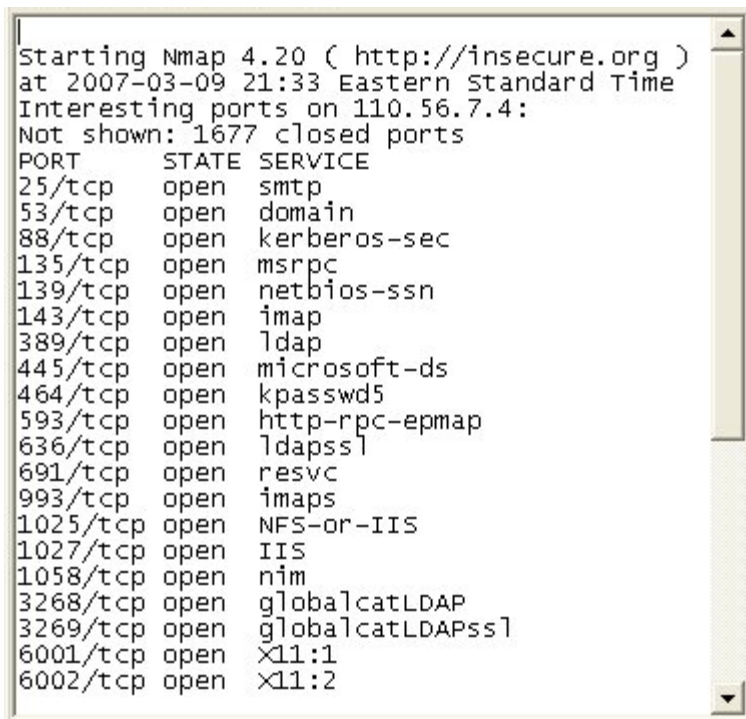
The security updates were being handled fairly well. There was an automated mechanism in place to distribute them. There was no clear policy to establish in what time frame systems should be updated, but in most cases, critical servers were updated within 24-48 hours and most workstations were updated within 5-7 days. More problematic was the fact that in spite of using an automated

distribution system, the defenders ability to know which systems had been updated and which had not been updated was sporadic and varied tremendously from site to site. During inspections, while somewhere around 80% of the systems tested had been updated within a week of the latest patch, there were also systems noted that were months out of date and on rare occasions, a system that was years out of date was discovered.

The basic reconnaissance already performed had produced a list of servers that were based on the edge of the perimeter defense. In some cases, the basic recon had provided enough information to allow for some kind of penetration. In other cases, it was simply a URL and an associated IP address.

More detailed reconnaissance was performed by running Nmap¹⁴ scans to identify ports that were open and sometimes the service that was running on the port. The Nmap scans were run with slow and stealthy settings to avoid attracting attention. The attackers had practiced this in the Academy against several different intrusion detection systems and knew what settings were needed to evade most default IDS settings. They also had a list of ports to avoid probing. This "hotlist" of ports was comprised mostly of ports related to classic known trojan programs (like SubSeven and BackOrifice2000) that would trigger an IDS response. An Nmap scan done with default settings will light up most intrusion detection consoles like a Christmas tree. As the hot ports are excluded and the speed and stealth settings are tinkered with, a port scan can be run without so much as a single IDS event being triggered. Through

their training in the Academy laboratory, the attackers had become quite proficient at these stealthy tactics. Since perimeter systems are usually buried under a snowstorm of malicious packets of all types, this was probably an unnecessary precaution, but the attackers decided to err on the side of safety.

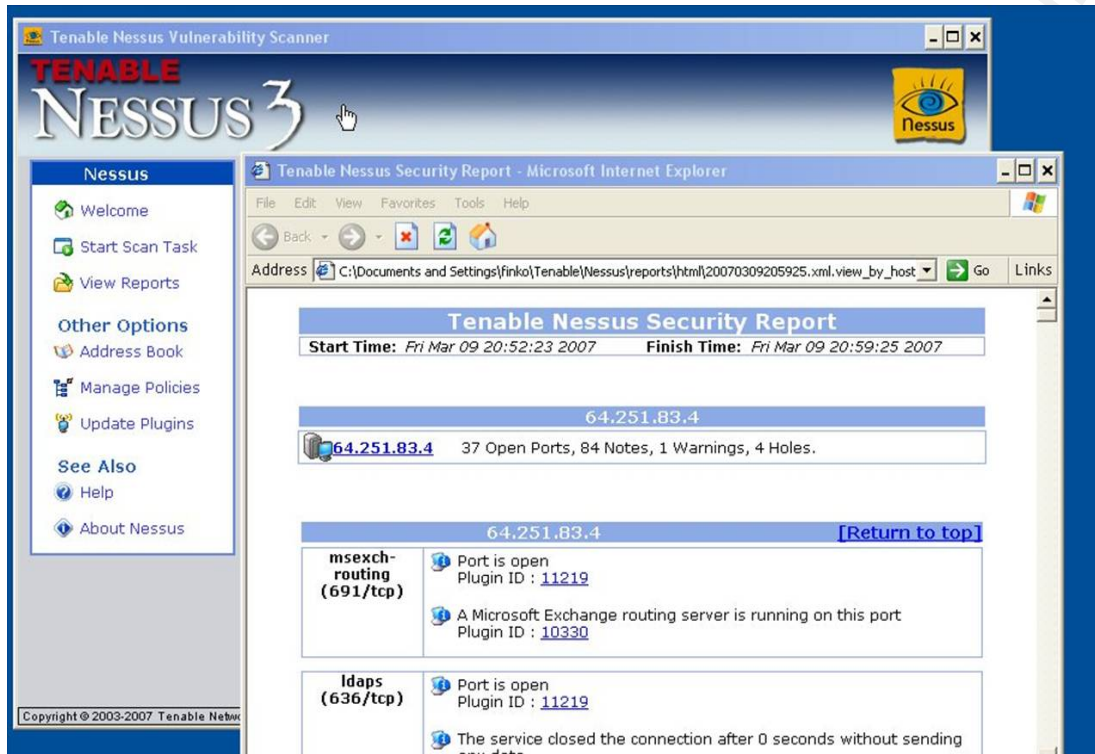


```
Starting Nmap 4.20 ( http://insecure.org )
at 2007-03-09 21:33 Eastern Standard Time
Interesting ports on 110.56.7.4:
Not shown: 1677 closed ports
PORT      STATE SERVICE
25/tcp    open  smtp
53/tcp    open  domain
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
143/tcp   open  imap
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
691/tcp   open  resvc
993/tcp   open  imaps
1025/tcp  open  NFS-or-IIS
1027/tcp  open  IIS
1058/tcp  open  nim
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
6001/tcp  open  X11:1
6002/tcp  open  X11:2
```

[a routine nmap port scan showing open ports in a laboratory environment]

A second layer of recon scanning was done using Nessus¹⁵ to identify vulnerabilities in the perimeter systems. While Nessus scans are usually quite noisy, again the team elected to do them as slowly and with options carefully selected to keep the chance of being noticed at a minimum. The map of ports open and services

running produced by nmap gave the attackers a reasonably good idea of what OS was running on the target system and how to tune the scan to be most effective.



[the front page of Nessus scan showing holes found in a server that may be exploitable – note: the IP address was on a private network for lab exercises]

The perimeter attack team had defined many security holes in the edge defenses that would allow penetration, but they were worried that exploits against these old vulnerabilities would be noticed by any intrusion detection defenses and so alert the defenses to their penetration efforts. The value of knowledge to be gained by measuring the response and assessing the viability of any intrusion detection capability was weighed carefully against the need for

caution and in the end they decided not to use the old exploits yet, but to wait until either a zero-day hole allowed them to slip through unnoticed, or one of the other teams had identified the defenses to a point where they knew they would not be detected. In the meantime, they continued probing whenever security updates were released (ones that did not offer penetration opportunities) to confirm the time frame in which the defenders normally did their patching and even identified a few sites that were slower than the rest. This could turn out to be very useful later.

5 - Wireless Network

A. DEFENDERS: AC-18 WIRELESS ACCESS RESTRICTIONS

[the italicized section below is a security control from NIST SP 800-53]

Control: The organization: (i) establishes usage restrictions and implementation guidance for wireless technologies; and (ii) documents, monitors, and controls wireless access to the information system. Appropriate organizational officials authorize the use of wireless technologies.

Guidance: NIST Special Publication 800-48 provides guidance on wireless network security with particular emphasis on the IEEE 802.11b and Bluetooth standards.

Control Enhancement 1: The organization uses authentication and encryption to protect wireless access to the information system.

[the following section is the PHA response to the control described above]

Implementation: See the Systems Level Controls Appendix for information regarding this control.

Systems Level Controls Appendix:

Section 27 - AC-18 Wireless Restrictions - see memorandum regarding "Wireless Activity in the PHA 123-456".

Memorandum: "Wireless Activity in the PHA 123-456"

This memorandum was not available.

The poorly written and vaguely defined defenders' policy examples in this paper are based on real life experiences. Despite the fact that the defenders' primary policy document regarding wireless restrictions seemed to be missing from the primary documentation, there was in fact another document that referenced NIST SP 800-48 and spelled out a complete framework for wireless security. Since this secondary document actually offered a semi-viable wireless defensive posture, it is offered below as the defenders' wireless policy.

- Encryption - WEP encryption must be turned on and must use 128 bit keys.
- SSIDs - SSIDs must be changed from defaults, must not reflect any information about the organization or location of the Access Point, and SSID cloaking must be turned on.
- MAC filtering - MAC address filtering must be used to restrict wireless access to only approved hardware addresses.
- Default settings - check all default settings, including administration password and access path, and change them or disable functions as needed. Turn off all access point services that are not being used (ftp, http, etc...).
- AP locations - place access points in interior positions and away from exterior walls and windows. Place them in secured locations to prevent unauthorized physical access.

B. ATTACKERS: KISMET

Kismet passively collects 802.11 wireless packets, usually by scanning across a range of channels, and presents information on the wireless networks it has seen. The packets are stored in tcpdump/wireshark format for later analysis and more detailed information is stored in several text files. Kismet tries to identify whether the traffic is encrypted or not and if so, by what means. It tries to identify the manufacturer of access points and clients that it sees. It collects client information that can be associated with an access point. It will tag "cloaked" access points and wait until it can correlate them with traffic from a client that reveals the SSID, effectively uncloaking them. It collects any IP addresses that are seen. It extracts Cisco equipment information from any CDP packets and stores it in a file. It can also collect GPS co-ordinates and save them for later mapping. Instead of scanning for traffic on all available channels, Kismet can be focused to collect traffic from only a specific channel or even a specific access point.

C. ATTACKERS: Aircrack

Aircrack is a suite of tools built around the primary tool called aircrack-ng which is designed to crack WEP encryption. It includes airodump-ng, which captures packets and does some analysis, and aireplay-ng, which performs packet replay injection to speed up cracking.

Airodump-ng can be used in conjunction with Kismet. Some of the functionality is redundant with Kismet, but one area where it is unique is in collection of the keys known as an "Initialization Vectors" (IVs) for cracking. The kismet packet dump file collects all packets and can grow quite large in a short amount of time. Airodump-ng can be set to collect only packets with IVs (ignoring all the beacon frames and other administrative packets), thus greatly reducing the size of the capture file. The airodump-ng IV capture file format is not compatible with Wireshark or most other packet reading tools, but it works fine with aircrack-ng. Airodump-ng also produces a nice list of access points and clients that it has seen along with their associated information. This data can easily be imported into a spreadsheet or database for further analysis.

Aircrack-ng is designed to use a hybrid blend of statistical analysis and FMS-style attacks against WEP encryption. It can also perform a dictionary attack against WPA encryption. It has the ability to read an IV collection file from airodump-ng at the same time as it is being collected. This is a very convenient feature. When a capture file is opened by Airodump-ng, a list of networks included in the packets is shown and the user is offered a choice of which network they wish to process for cracking.

Aireplay-ng is a tool that can replay a packet and inject it back into a wireless data stream, usually for the purpose of producing more IVs in response and greatly accelerating the cracking process, which depends on how many IVs have been collected.

For high volume wireless networks that produce a lot of IV containing packets in a short time, airodump-ng can be used to collect about 150,000 packets (passively) and then aircrack-ng can be used to crack them and produce the key needed to decode the traffic completely.

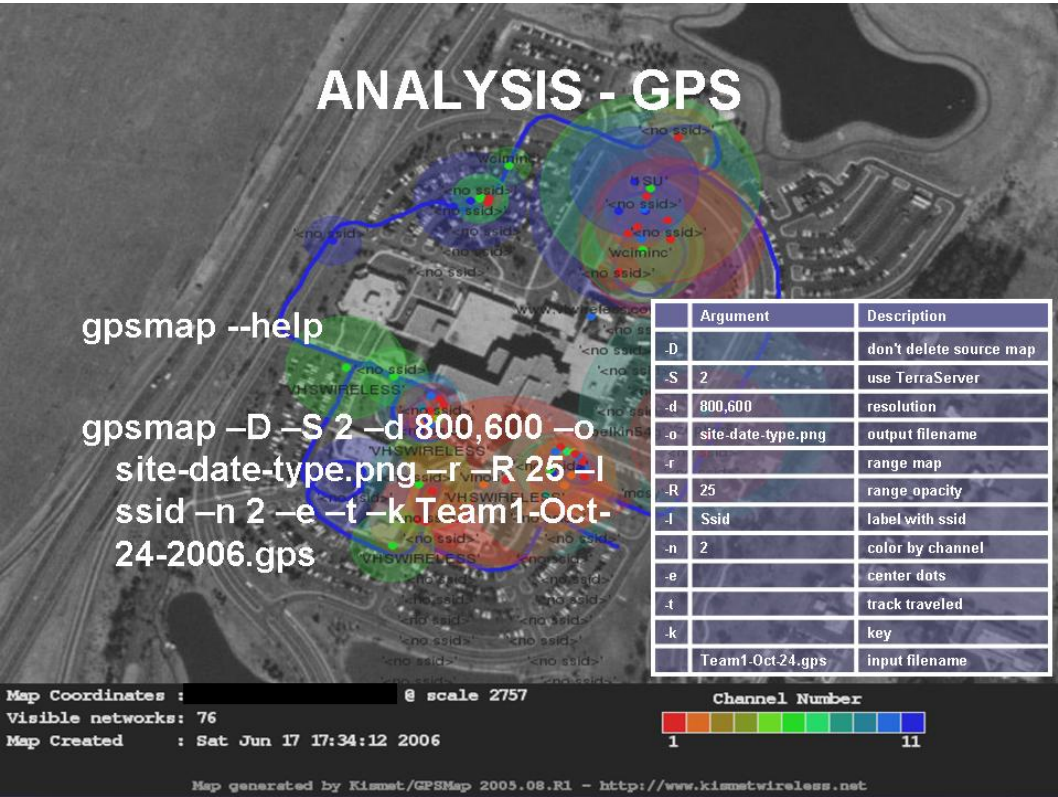
In a situation requiring haste and no stealth, it is possible to use aireplay-ng to speed up the collection process and accomplish cracking a 128 bit WEP key in well under ten minutes. However, the injection process used by aireplay-ng is noisy, intrusive, probably illegal, and should be picked up by any wireless intrusion detection process, if one is being used. In most wireless installations today, a wireless IDS system is not being used.

D. Scenario (Wireless Network)

The attackers used Kismet to survey the wireless landscape and then performed analysis on the data they collected. From the initial reconnaissance phase, they already knew that the facilities were using both 802.11b and 802.11g band with WEP encryption. A preliminary boundary survey around the geographic perimeter of each target facility helped by defining what wireless activity was originating inside the perimeter and what was found outside the border. In most cases, this was done by war-driving (in a moving vehicle) and by either cruising slowly around the edge of the facility perimeter or parking to collect packets then moving and repeating the process. They did a lot of crisscrossing and retracing of the same routes in order to fill in the data set completely. They spread their efforts out over many days in order to work slowly and

not attract attention, but also in order to make sure they saw all the wireless activity that was available and were not limited to a single snapshot in time.

Running the GPS data file through a mapping application called gpsmap produced a map of the facility perimeter that showed the locations of the access points found. The locations are approximated by the data co-ordinates collected, so if the collection point (car) was moving in a straight line, the location might not be very accurate. But with the collectors deliberately creating a grid-like driving pattern, the locations can be trusted as fairly accurate. The map made it easy to identify wireless sources that are outside the target perimeter and screen them out from future collections. When the location of an access point seemed ambiguous, a YAGI (directional) antenna was used to pinpoint the source.

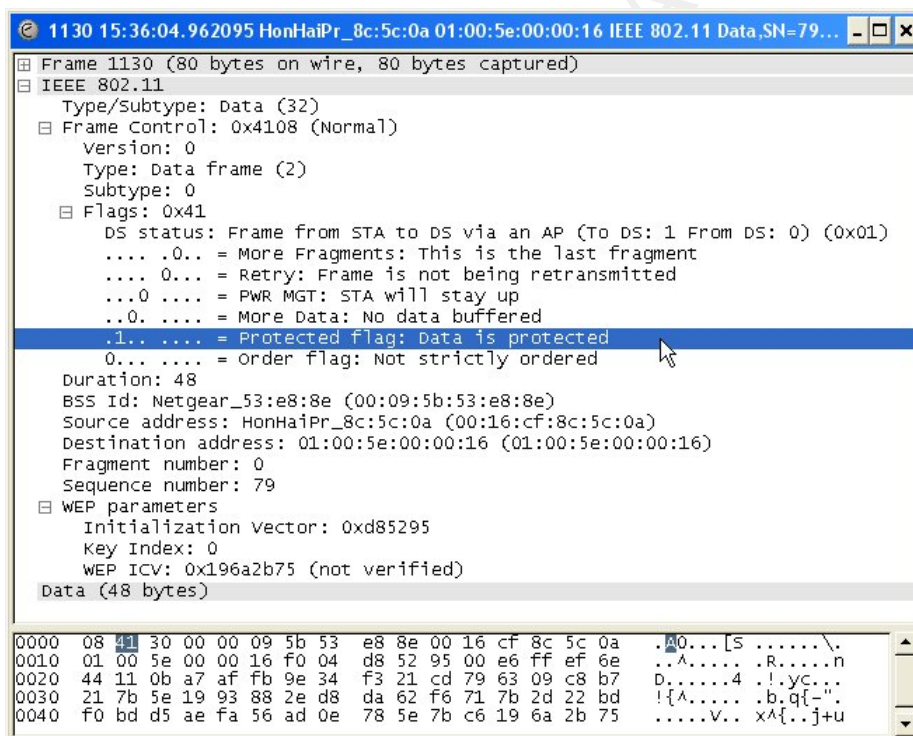


[example of a gpsmap produced from Kismet data]

Further analysis was done by importing the .csv file into Excel, and sorting and color coding entries by MAC address, SSID, encryption type and more. In large installations, access points are often purchased in large quantities and this can be observed by noting MAC addresses that are nearly sequential. MAC addresses consist of a six character prefix that represents the manufacturer and a six character suffix that is essentially the same thing as a serial number. When the suffixes of a group of MAC addresses are sequential or nearly sequential, the observer can make a good guess that they were purchased in a batch lot and are all deployed by the same organization.

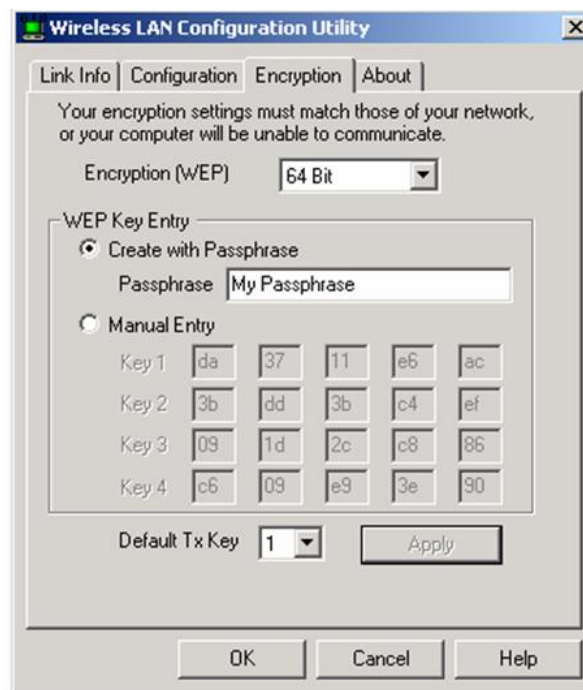
Once the boundary of the wireless zone was been defined, more concentrated collection of the interior began. At some points, Kismet was locked onto a single channel and at other times it was set to filter out traffic related to a single BSSID (MAC address of an access point).

As the collected data was analyzed, the fact that WEP encryption was being used was confirmed by looking at the privacy bit in the frame control field found in management frames. The privacy bit by itself only indicates encryption of some form, but the following WEP parameters confirm that the privacy technique being used is in fact WEP.



[high-lighted row shows privacy bit - WEP parameters appear near the bottom of the packet]

The length of the key cannot be determined by observation alone and must be deduced by cracking efforts. In order to eliminate the easiest methods first, the attackers ran a utility called "wep_crack" which can crack the "Nessus Datacom" vulnerability in shorter 64 bit keys and often yields results in less than one second with only two packets.

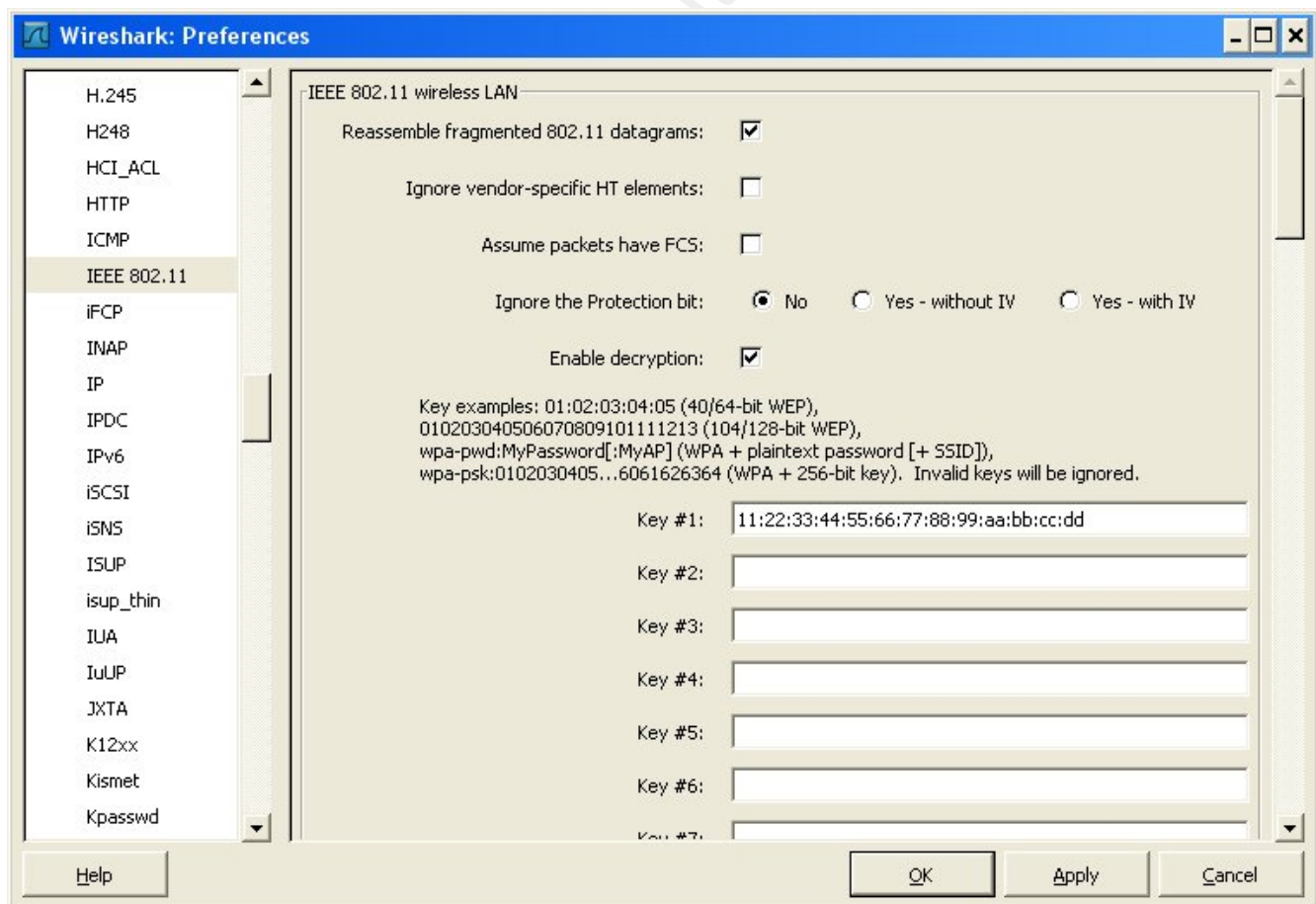


[a commonly seen configuration utility that uses the algorithm vulnerable to the Nessus Datacom attack]

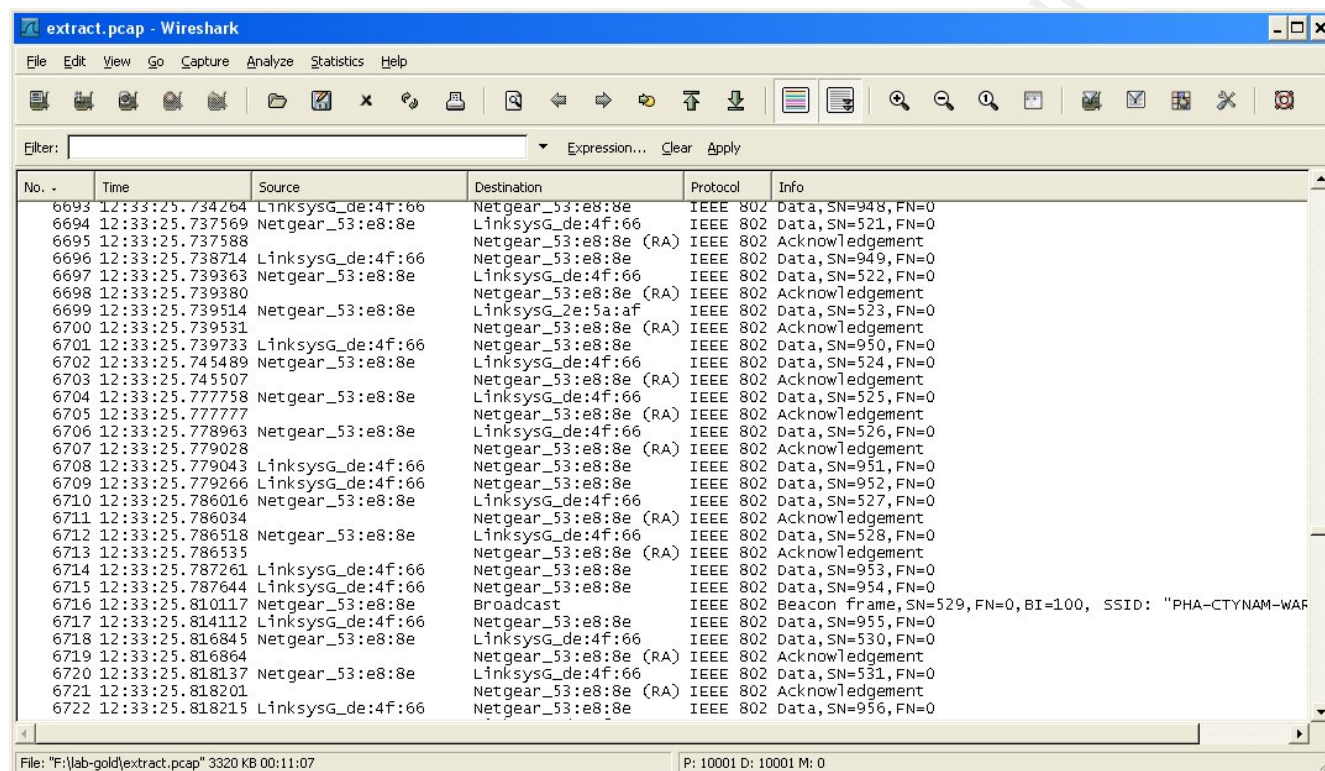
None of the traffic collected produced results with this method. The attackers next ran a tool called "wep_attack" which performs a

dictionary attack against packets encrypted with both 64 bit and 128 bit keys, but also produced no results with this. It was time to bring the larger guns into play, so they started airodump-ng on collecting packets from a specific access point. Once about 150,000 packets had been collected from the access point, they were fed into aircrack-ng and within a few minutes, the WEP key was produced.

Once the encryption key has been retrieved, it can be put into Wireshark and used to decode all the encrypted packets for further analysis. All visible traffic on the wireless network can now be read in plain text.



Screenshot - adding key into wireshark to decrypt packets



Screenshot - packets in wireshark as they are collected

| No. | Time | Source | Destination | Protocol | Info |
|------|-----------------|-----------------------|-----------------------|----------|--|
| 6693 | 12:33:25.734204 | 192.168.1.2 | 216.239.51.104 | TCP | [TCP window update] 1115 > http [ACK] Seq=1016 Ack=120 |
| 6694 | 12:33:25.737569 | 216.239.51.104 | 192.168.1.2 | TCP | [TCP segment of a reassembled PDU] |
| 6695 | 12:33:25.737588 | 192.168.1.2 | Netgear_53:e8:8e (RA) | IEEE 802 | Acknowledgement |
| 6696 | 12:33:25.738714 | 192.168.1.2 | 216.239.51.104 | TCP | 1115 > http [ACK] Seq=1016 Ack=14944 Win=14660 Len=0 |
| 6697 | 12:33:25.739363 | 216.239.51.104 | 192.168.1.2 | TCP | [TCP segment of a reassembled PDU] |
| 6698 | 12:33:25.739380 | Netgear_53:e8:8e (RA) | Netgear_53:e8:8e (RA) | IEEE 802 | Acknowledgement |
| 6699 | 12:33:25.739514 | 168.95.1.1 | 192.168.1.3 | ICMP | Echo (ping) reply |
| 6700 | 12:33:25.739531 | Netgear_53:e8:8e (RA) | Netgear_53:e8:8e (RA) | IEEE 802 | Acknowledgement |
| 6701 | 12:33:25.739733 | 192.168.1.2 | 216.239.51.104 | TCP | [TCP window update] 1115 > http [ACK] Seq=1016 Ack=149 |
| 6702 | 12:33:25.745489 | 216.239.51.104 | 192.168.1.2 | TCP | http > 1116 [ACK] Seq=1078 Ack=516 win=16080 Len=0 |
| 6703 | 12:33:25.745507 | Netgear_53:e8:8e (RA) | Netgear_53:e8:8e (RA) | IEEE 802 | Acknowledgement |
| 6704 | 12:33:25.777758 | 216.239.51.104 | 192.168.1.2 | TCP | [TCP segment of a reassembled PDU] |
| 6705 | 12:33:25.777777 | Netgear_53:e8:8e (RA) | Netgear_53:e8:8e (RA) | IEEE 802 | Acknowledgement |
| 6706 | 12:33:25.778963 | 216.239.51.104 | 192.168.1.2 | TCP | [TCP segment of a reassembled PDU] |
| 6707 | 12:33:25.779028 | Netgear_53:e8:8e (RA) | Netgear_53:e8:8e (RA) | IEEE 802 | Acknowledgement |
| 6708 | 12:33:25.779043 | 192.168.1.2 | 216.239.51.104 | TCP | 1115 > http [ACK] Seq=1016 Ack=17804 Win=14660 Len=0 |
| 6709 | 12:33:25.779266 | 192.168.1.2 | 216.239.51.104 | TCP | [TCP window update] 1115 > http [ACK] Seq=1016 Ack=178 |
| 6710 | 12:33:25.786016 | 216.239.51.104 | 192.168.1.2 | TCP | [TCP segment of a reassembled PDU] |
| 6711 | 12:33:25.786034 | Netgear_53:e8:8e (RA) | Netgear_53:e8:8e (RA) | IEEE 802 | Acknowledgement |
| 6712 | 12:33:25.786518 | 216.239.51.104 | 192.168.1.2 | HTTP | HTTP/1.1 200 OK (JPEG JFIF image) |
| 6713 | 12:33:25.786535 | Netgear_53:e8:8e (RA) | Netgear_53:e8:8e (RA) | IEEE 802 | Acknowledgement |
| 6714 | 12:33:25.787261 | 192.168.1.2 | 216.239.51.104 | TCP | 1115 > http [ACK] Seq=1016 Ack=20664 Win=14660 Len=0 |
| 6715 | 12:33:25.787644 | 192.168.1.2 | 216.239.51.104 | TCP | [TCP window update] 1115 > http [ACK] Seq=1016 Ack=206 |
| 6716 | 12:33:25.810117 | Netgear_53:e8:8e | Broadcast | IEEE 802 | Beacon frame, SN=529, FN=0, BI=100, SSID: "PHA-CTYNAM-WAR" |
| 6717 | 12:33:25.814112 | 192.168.1.2 | 216.239.51.104 | HTTP | GET /ThumbnailServer2?app=vss&contentid=25cb8dfb0f8a50 |
| 6718 | 12:33:25.816845 | 216.239.51.104 | 192.168.1.2 | TCP | [TCP segment of a reassembled PDU] |
| 6719 | 12:33:25.816864 | Netgear_53:e8:8e (RA) | Netgear_53:e8:8e (RA) | IEEE 802 | Acknowledgement |
| 6720 | 12:33:25.818137 | 216.239.51.104 | 192.168.1.2 | TCP | [TCP segment of a reassembled PDU] |
| 6721 | 12:33:25.818201 | Netgear_53:e8:8e (RA) | Netgear_53:e8:8e (RA) | IEEE 802 | Acknowledgement |
| 6722 | 12:33:25.818215 | 192.168.1.2 | 216.239.51.104 | TCP | 1116 > http [ACK] Seq=516 Ack=2508 Win=17520 Len=0 |

Screenshot – the same packets in wireshark after the encryption key has been added to decrypt them

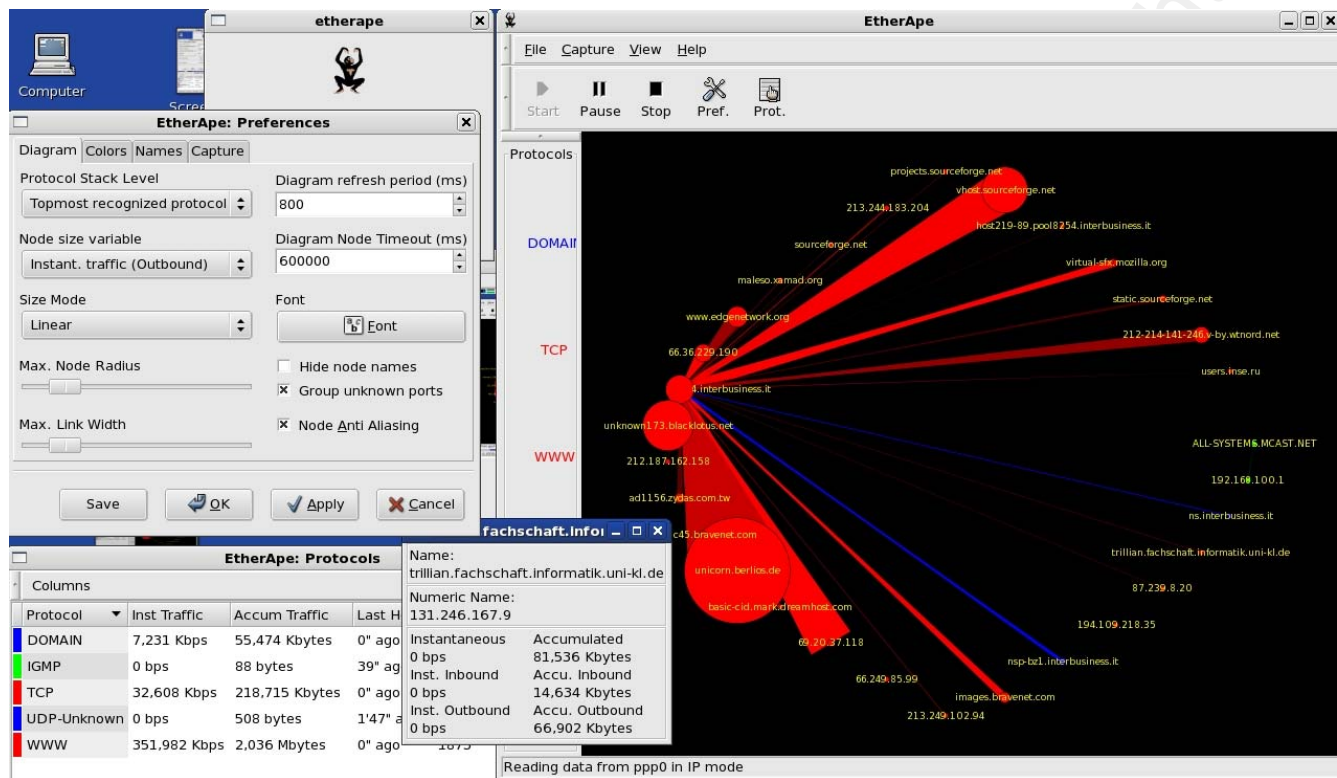
The key can also be used to configure the wireless connection and associate with an access point and become part of the target network. At this point, with the encryption key available, once a live connection is established, all future network traffic seen by the client will be decrypted using the key and can be observed by wireshark just like normal network packets.

In most cases, the defenders were using MAC address filtering as a wireless defense as their policy had stated. Even with the encryption key, this filtering can prevent an unauthorized connection

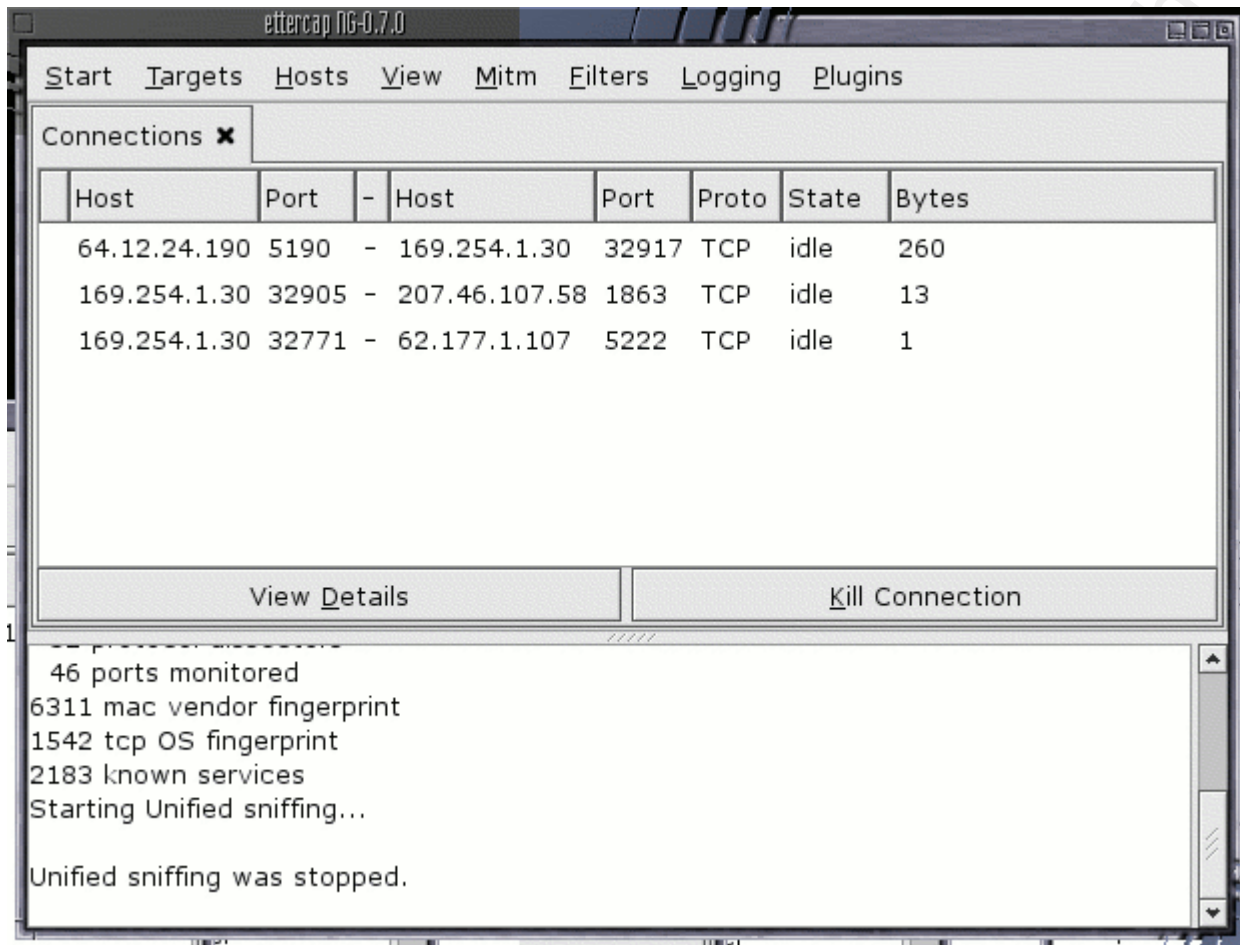
from being completed. Earlier analysis had produced a list of MAC addresses of clients that had been observed connecting to access points. A valid MAC address was selected from the list and after checking to make sure that it was not currently active, the attacking system was configured to use that MAC address. With both a cloned MAC address and the encryption key, the attacking system could connect with the wireless network.

Because of the obtrusive nature of aireplay's packet injection technique, the attackers' decided to not use it at least for their early penetration attempts. They didn't want to risk being noticed and the high volume of traffic available made it unnecessary to use.

The attackers now engaged in passive collection of network packets and analysis of whatever they were able to collect. Tools such as Etherape, Ettercap-NG and p0f can be used either online in passive mode (putting out no packets at all, making them invisible) or simply by reading in packet capture files while offline. They can identify the source and target of traffic flow, show ports and protocols being used, identify the OSes, capture text information and more.



[example of Etherape - from <http://etherape.sourceforge.net/images/>]



[example of Ettercap - from
<http://ettercap.sourceforge.net/screenshots.php>]

Eventually, a decision point was reached on whether or not to press the attack further, which would require some form of action that might be detected, instead of silent passive collection. The wireless attack team had successfully penetrated the perimeter and now had a presence inside the defenses that allowed them to see far more of the network traffic than the defenders would have believed possible, but they decided to stop here and wait for a zero-day vulnerability before pushing the penetration any deeper. In the

meantime, they continued to passively collect network data, analyze it and add it into their ever growing database of PHA network information.

6 - Bypass

A. DEFENDERS: SI-3 MALICIOUS CODE PROTECTION

[the italicized section below is a security control from NIST SP 800-53]

Control: The information system implements malicious code protection that includes a capability for automatic updates.

Guidance: The organization employs virus protection mechanisms at critical information system entry and exit points (e.g., firewalls, electronic mail servers, remote-access servers) and at workstations, servers, or mobile computing devices on the network. The organization uses the virus protection mechanisms to detect and eradicate malicious code (e.g., viruses, worms, Trojan horses) transported: (i) by electronic mail, electronic mail attachments, Internet accesses, removable media (e.g., diskettes or compact disks), or other common means; or (ii) by exploiting information system vulnerabilities. The organization updates virus protection mechanisms (including the latest virus definitions) whenever new releases are available in accordance with organizational configuration management policy and procedures. Consideration is given to using virus protection software products from multiple vendors (e.g., using one vendor for boundary devices and servers and another vendor for workstations).

Control Enhancement 1: The organization centrally manages virus protection mechanisms.

Control Enhancement 2: The information system automatically updates virus protection mechanisms.

[the following section is the PHA response to the control described above]

Implementation: All PHA computer systems must have anti-virus protection software installed, active and up to date. Automation of the update process is critical and will be done from a central location. Anti-virus signature files (also known as .DAT files) must be updated on a periodic basis or whenever a new threat materializes.

B. ATTACKERS: Custom Malware

Anti-virus software and most intrusion detection tools do a lot of their threat detection by recognizing previously identified signatures. A signature is derived from some of the code that the virus or trojan program is likely to depend upon and not very likely to be modified without altering the functionality of the malware. Malware is collected and analyzed to produce the signatures when it is noticed and that most often occurs when it has become widespread and created an effect. If the malware is not widely used and goes un-noticed and uncollected, it won't be analyzed for development of a detection signature.

Targeted attacks, designed to be used against a single target, can avoid signature detection. Since the malware is custom designed to avoid any known signatures and has never been widely released, a signature for it will not exist and no signature detection mechanism will find it, whether in anti-virus software, intrusion detection software, or any other form. Malware can also be disguised from signature detection by using polymorphic tools that change the code constantly, creating a unique version with a unique signature each time the program is created. Polymorphic toolkits such as: ADMutate, PHATBOT, Jujuskins, TAPion and CLET put this kind of

functionality within the reach of the average skilled malware creator, if not the novice. As polymorphic shell code has become more common, defenses have adapted to detect it. More advanced detection tools use traffic profiling techniques, known as "anomaly detection" to filter out traffic that does not fit a profile of normal traffic. This takes malware detection a step beyond merely using signatures and might detect a threat that has no known signature. An offensive counter to anomaly detection is described in a paper titled, "Polymorphic Blending Attacks".

Joanna Rutkowska describes her concept of hard to find malware as, "Type II: Malware which modifies things which are designed to be modified (DATA sections)."¹⁶ One of the examples she offers of type II malware includes the FU rootkit by Jamie Butler. An FU based module has been available for some time for the old classic backdoor program, BackOrifice2000. FU-like features have turned up in Rbot and the Myfip worm. Although this class of malware is more difficult to detect, it can be discovered. Joanna goes on to describe an even more stealthy class of malware which she calls "stealth by design". In this "proof of concept" design, the malware has its' own shell and TCP/IP stack, minimizing traceability.

In another separate, but real-life example of stealthy malware, the Gozi trojan existed in the wild for over fifty days in the beginning of 2007, and it has been estimated that the first variant of it infected more than 5,000 hosts and stole account information for over 10,000 users. Gozi's primary function was to steal credentials being sent over SSL connections before they were

encrypted and add them to a database server that would dispense them on demand in exchange for payment. Had the malware author made a better choice of the packing utility used, the trojan may have gone much longer before being detected.

C. Scenario (Bypass)

The by-pass attack team used several custom made trojan programs to completely by-pass the perimeter defenses. The trojan programs were delivered by email and web pages, but the anti-virus defenses were not triggered because the trojans did not have any known signatures. Many different versions were used in the hopes that if one was noticed, collected and scrutinized; it would not reveal information that could create a signature that would detect the others.

Email addresses for people inside the target organization were collected during the basic reconnaissance and more were accumulated over time with searches based on domains and user names. Some emails simply sent attachments, often with a spoofed source address, designed to convince the recipient that the email was valid, but not allow any trace back to the real sender. Other emails sent links to web pages that had downloads available or malicious scripts to run. In either case, the trojans were made to look like a valid object and usually delivered some kind of camouflaging action while the program was being installed to convince the user that nothing was amiss.

Once the trojan program was installed, it activated a remote access backdoor to allow the attackers to control the compromised system. The backdoors used a variety of techniques to get out past the perimeter. The goal was to allow responses and data from the compromised host inside the perimeter to get outside to a command and control node and to allow more instructions to get back inside to the compromised host. Web traffic was most often the carrier for this, since the perimeter defenses were required to allow it to pass through. Sometimes the web traffic was encrypted with SSL. In some cases, encrypted secure shell sessions were used. In other cases, Email traffic was used to carry embedded data in attached files, usually encrypted.

None of this activity was detected by the defenders. Once the attackers had established a presence on a system, they followed up with variations of the techniques described in Chapter 8 - Entrench.

7 - Walk-in

A. DEFENDERS: PE-3 PHYSICAL ACCESS CONTROL

[the italicized section below is a security control from NIST SP 800-53]

Control: The organization controls all physical access points (including designated entry/exit points) to facilities containing information systems (except for those areas within the facilities officially designated as publicly accessible) and verifies individual access authorizations before granting access to the facilities. The organization also controls access to areas officially designated as publicly accessible, as appropriate, in accordance with the organization's assessment of risk.

Guidance: The organization uses physical access devices (e.g., keys, locks, combinations, card readers) and/or guards to control entry to facilities containing information systems. The organization secures keys, combinations, and other access devices and inventories those devices regularly. The organization changes combinations and keys: (i) periodically; and (ii) when keys are lost, combinations are compromised, or individuals are transferred or terminated. After an emergency-related event, the organization restricts reentry to facilities to authorized individuals only. Workstations and associated peripherals connected to (and part of) an organizational information system may be located in areas designated as publicly accessible with access to such devices being appropriately controlled.

[the following is the PHA response to the security control described above]

Implementation: Physical controls include: locks and keys, combinations, badge access controls systems, guards, raised computer room floors, uninterruptible power supplies, smoke detectors, alarm systems, sprinkler systems, fire extinguishers, air conditioning systems and more.

Note - the defenders seem to have mixed in some environmental controls with the physical controls, but at least they are from the same family.

B. DEFENDERS: AC-11 SESSION LOCK

[the italicized section below is a security control from NIST SP 800-53]

Control: The information system prevents further access to the system by initiating a session lock that remains in effect until the user reestablishes access using appropriate identification and authentication procedures.

Guidance: Users can directly initiate session lock mechanisms. The information system also activates session lock mechanisms automatically after a specified period of inactivity defined by the organization. A session lock is not a substitute for logging out of the information system.

[the following is the PHA response to the security control described above]

Implementation: The system will use a 15 minute timeout to initiate a session lock with the Windows screensaver mechanism.

This will be consistently applied across the organization by using global policy settings.

NOTE - A corollary to this control is AC-12 SESSION TERMINATION which sets another timeout factor for ending an unattended session.

C. DEFENDERS: AC-6 LEAST PRIVILEGE

[the italicized section below is a security control from NIST SP 800-53]

Control: The information system enforces the most restrictive set of rights/privileges or accesses needed by users (or processes acting on behalf of users) for the performance of specified tasks.

Guidance: The organization employs the concept of least privilege for specific duties and information systems (including specific ports, protocols, and services) in accordance with risk assessments as necessary to adequately mitigate risk to organizational operations, organizational assets, and individuals.

[the following is the PHA response to the security control described above]

Implementation: PHA policy directs the IT Manager for the site to establish controls that separate duties to ensure least privilege and establish accountability. This process must be monitored and periodically updated.

D. Scenario (Walk-in)

Most public hospitals are open to the public and this creates special security issues. For the third vector, the attackers simply walked into the hospital and proceeded to penetrate network systems in a variety of ways. From previous experience, they knew that physicians' work rooms pose a serious risk in most hospitals. Most

hospitals provide their doctors with some kind of work room in which they can access the internet to search for medical research information and print it out. The typical work room has one or two networked printers in it and four to six workstations with network access. There is usually no security at the door to the workroom. The doctors walk in and sit down at the workstation and do their work, then leave and often leave the workstation logged in with their credentials.

The attackers' basic plan was to walk into the hospital, proceed to the physicians' work room, which had been located either by the Google general reconnaissance collection or by a previous walk around reconnaissance, and sit down at a system that had been left logged in by a doctor. In some cases, they used fake ID badges that matched the ones used by the facility; in other cases they included a white overcoat like the doctors wear. In none of the intrusion attempts was any attacker ever challenged or even spoken to, and in all facilities, the intrusions were successful. In about half of the attempts, a logged in workstation was available immediately, but in the rest of the attempts, it only took a 5-10 minute wait until a doctor left a workstation that was still logged in. The attackers reported that it was very rare to notice a doctor actually logging out of a workstation after using it.

Once the attacker was seated at a logged in workstation, they would usually insert a USB flash drive into a USB socket and begin executing tools from it. The first step was to run an information collecting batch file that dumped system and network information back to the flash drive in the form of text files. While the batch was running, the attacker would determine if the user account had

administrative privileges by right clicking on the start button. If the popup window included "Open All Users" and "Explore All Users" the account had admin rights. If the account had admin rights, the password hash dumping batch file would also be run. In either case, the next step was to install a root kit and a collection of tools that would be hidden inside it. All of this activity normally took less than three minutes and often it was possible to move to another vacant workstation and repeat the process.

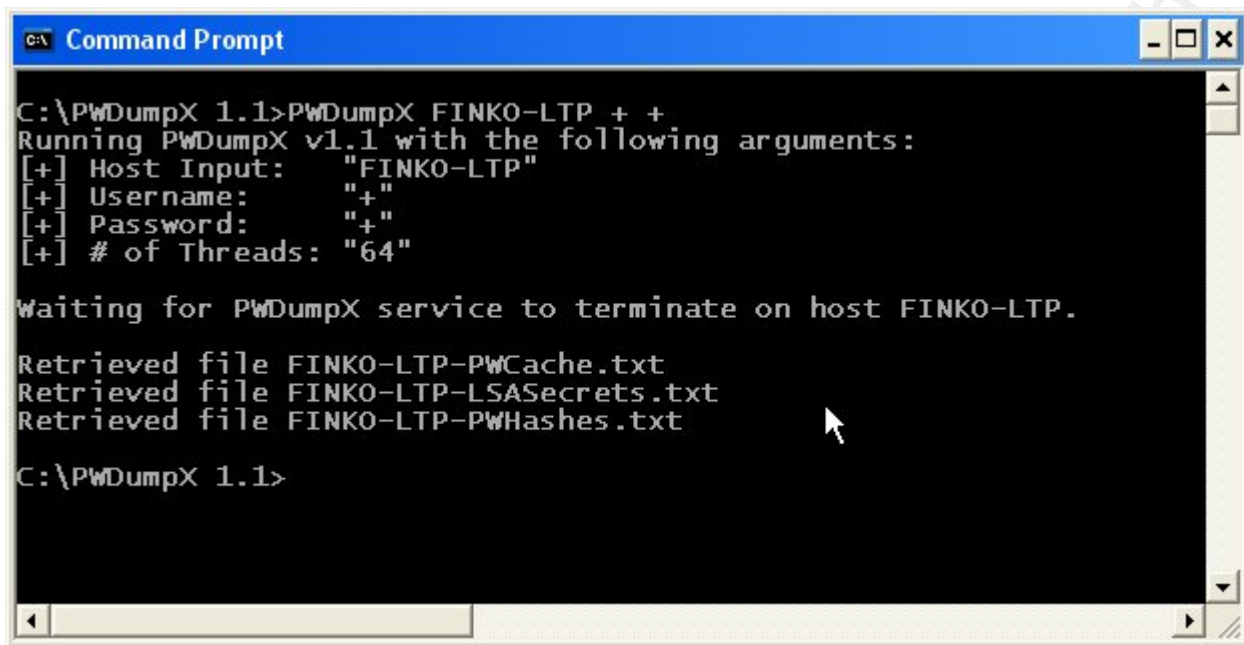
If the physician's work room was too crowded or the attackers felt they had exhausted its potential, they would move on to other systems. There were also systems available in some public waiting rooms and even unattended systems in various parts of the hospital. It would be much riskier to sit down at a computer in some place like a nursing station, unless a good cover story was ready to be used, such as a technical support procedure or system updates that needed to be run. But in most cases, the attackers resorted to leaving a flash drive loaded with their tools just sitting nearby and counted on the hospital personnel to put it into the system for them. They were pre-configured to auto-play a program that showed the user some flash screens about hospital administration while it loaded the root kit and tools in the background, then deleted most traces of its activity, including scrubbing the tool files from the flash drive, so that any forensic investigation would not produce much.

The attackers also carried with them several hardware key-logger devices to be installed at the end of a keyboard cable. The key-logger is similar in size and appearance to the plug at the end of

the keyboard cable and is installed by simply pulling the keyboard cable out of the computer, inserting the key-logger between the cable plug and the socket on the computer and reconnecting the keyboard. The process takes a few seconds. The key-logger records every keystroke typed on the keyboard and it can be dumped later. The attackers planned on returning in a few days to retrieve the key-loggers and the data they held.

Most of the accounts that were used for access did not have admin rights, but eventually the attackers would find one that did, and then would dump the password hashes for later cracking. Pwdump is a utility that can extract password hashes from a windows system. It requires admin access to run. The current version is pwdump6 and it comes with a "wrapper" program called fgdump that makes it work more effectively. The fgdump wrapper adds the ability to stop then restart anti-virus software and also adds a cachedump tool. When you run pwdump successfully, it produces a text file output of the hashes that can be imported into most password cracking tools.

The attackers had already determined from the earlier google reconnaissance what brand of anti-virus software the defenders used and knew that it would recognize and automatically quarantine the pwdump.exe program normally used to extract and dump password hashes. They had tracked down an alternative version, PWDumpX.exe¹⁷ and tested it in their lab against current anti-virus DAT files. Since the alternative version was a re-write of the more widely known code in the original pwdump, the existing signatures did not recognize it.



```
C:\PWDumpX 1.1>PWDumpX FINKO-LTP + +
Running PWDumpX v1.1 with the following arguments:
[+] Host Input: "FINKO-LTP"
[+] Username: "+"
[+] Password: "+"
[+] # of Threads: "64"

Waiting for PWDumpX service to terminate on host FINKO-LTP.

Retrieved file FINKO-LTP-PWCache.txt
Retrieved file FINKO-LTP-LSASecrets.txt
Retrieved file FINKO-LTP-PWHashes.txt

C:\PWDumpX 1.1>
```

[PWDumpX running - note - the attackers lab could have easily produced many such variants themselves]

The walk-in attack team had also volunteered to help the wireless attack team by installing some rogue access points. When they found a network port that would respond without requiring any authentication (and that was nearly all of the ports tested) they plugged in a wireless access point and turned it on. The team had debated whether or not to attempt to conceal the equipment by taping it underneath tables, but in the end decided to simply leave them out in plain site, under the assumption that most people would not touch a piece of computer equipment that they knew nothing about. This strategy apparently worked, because all of the rogue access points remained in operation throughout the penetration.

A variety of simple consumer equipment was used, so that each rogue had a different appearance, but they were all selected for use because the wireless radio card inside could either be set to extra channels beyond the standard eleven channels licensed for use in the U.S. or they could be flash updated to accommodate that feature. In Europe and Japan, channels 12 and 13 are also allowed and in Japan only, channel 14 is allowed. The rogues were all set to channel 14 to make it more difficult for any rogue hunting defenders to find them.

8 - Entrench

A. DEFENDERS: IA-2 USER IDENTIFICATION AND AUTHENTICATION

[the italicized section below is a security control from NIST SP 800-53]

Control The information system uniquely identifies and authenticates users (or processes acting on behalf of users)

Guidance: Authentication of user identities is accomplished through the use of passwords, tokens, biometrics, or in the case of multifactor authentication, some combination therein. FIPS 201 and Special Publications 800-73 and 800-76 specify a personal identity verification (PIV) card token for use in the unique identification and authentication of federal employees and contractors. NIST Special Publication 800-63 provides guidance on remote electronic authentication. For other than remote situations, when users identify and authenticate to information systems within a specified security perimeter which is considered to offer sufficient protection, NIST Special Publication 800-63 guidance should be applied as follows: (i) for low-impact information systems, tokens that meet Level 1, 2, 3, or 4 requirements are acceptable; (ii) for moderate-impact information systems, tokens that meet Level 2, 3, or 4 requirements are acceptable; and (iii) for high-impact information systems, tokens that meet Level 3 or 4 requirements are acceptable. In addition to identifying and authenticating users at the information system level, identification and authentication mechanisms are employed at the application level, when necessary, to provide increased information security for the organization.

Control Enhancement 1: The information system employs multifactor authentication.

[the following is the PHA response to the security control described above]

Implementation: The minimum password length is to be set to 8 characters. Password complexity requirements must be set to "enabled" and requires three out of four factors in each password, including: lower case letters, upper case letters, numbers and special characters.

NOTE - the password length and complexity settings were incorrectly included in AC-2 Account Management, but have been presented here nonetheless.

B. ATTACKERS: A Simple Batch File

Commands issued from a windows command prompt can be used to collect a lot of system information. Some of the data collected includes:

- User accounts
- Share names
- Make, model and hardware configuration
- Operating system specifics
- Network adapter configuration (IP address, MAC address and many more)
- DNS server addresses

- Admin accounts
- Live network connections
- Security patch status
- Processes that are running (can reveal defenses such as Anti-Virus or HIPS and possibly weaknesses/attack vectors)
- The configuration of services on the system
- External share connections

Here is a sample batch file which uses this technique to capture all of this information and more, generally in less than two minutes:

```
echo off
echo collecting basic information
net users > %userdomain%-%computername%-netusers.txt
net accounts > %userdomain%-%computername%-netaccounts.txt
net localgroup > %userdomain%-%computername%-netlocalgroup.txt
net localgroup administrators > %userdomain%-%computername%-netlocalgroupadmins.txt

echo Collecting system information...
systeminfo > %userdomain%-%computername%-systeminfo.txt

echo Collecting ipconfig information...
ipconfig /all > %userdomain%-%computername%-ipconfig.txt

echo Collecting netsh diag information...
netsh diag show all /v > %userdomain%-%computername%-netshdiag.txt

echo Collecting net stats information...
net time > %userdomain%-%computername%-netstats.txt
net user >> %userdomain%-%computername%-netstats.txt
net share >> %userdomain%-%computername%-netstats.txt
net session >> %userdomain%-%computername%-netstats.txt
net statistics workstation >> %userdomain%-%computername%-netstats.txt
net statistics server >> %userdomain%-%computername%-netstats.txt
netstat -ano >> %userdomain%-%computername%-netstats.txt

echo Collecting task and service information...
tasklist > %userdomain%-%computername%-tasks.txt
sc query > %userdomain%-%computername%-services.txt
```



```
echo Collecting audit information...  
auditpol > %userdomain%-%computername%-audit.txt
```

[note - this last command depends on the auditpol.exe program being present - it may be available in a resource kit, or built into the operating system, or it can be included with the attackers' tools.]

WMIC stands for "Windows Management Instrumentation Command", and can be used to both read configuration information and write changes to both local and remote systems. WMIC is found on XP, Windows 2003 and Vista, but can also be used to read and manage Windows 2000 systems. It requires administrative privileges.

You can run WMIC in interactive mode by entering "wmic" at a command prompt. You will find yourself at a "wmic:root\cli" prompt and can enter commands. To run WMIC in non-interactive mode, simply type "wmic" followed by whatever parameters you wish, from a command prompt. This is useful for batch operations. For help, type "/" from a wmic prompt or "wmic /?" from a normal command prompt to see a list of switches and options available.

"Aliases" are used to reference WMI classes and they are listed in help. One such alias is "computersystem". Entering the command "wmic computersystem" will show a list of information about the system. The verb "list" is the default and is implied when not specified, so "wmic computersystem list" offers the same output. This can be modified with adverbs to show "wmic computersystem list brief" or "wmic computersystem list full". There are also output formatting options.

Here is a batch file which uses WMIC commands to collect system information:

```
echo off
echo collecting system information...
wmic /output:"%userdomain%-%computername%-computersystem.txt" computersystem list
/format:table
echo ...computersystem done

wmic /output:"%userdomain%-%computername%-os.txt" os list full /format:table
echo ...os done

wmic /output:"%userdomain%-%computername%-environment.txt" environment list brief
/format:table
echo ...environment done

wmic /output:"%userdomain%-%computername%-process.txt" process list brief
/format:table
echo ...process done

wmic /output:"%userdomain%-%computername%-sysaccount.txt" sysaccount list full
/format:table
echo ...sysaccount done

wmic /output:"%userdomain%-%computername%-service.txt" service list full
/format:table
echo ...service done

echo collecting patch information...
wmic /output:"%userdomain%-%computername%-qfe.txt" qfe list full /format:table
echo ...qfe done

echo collecting network information...
wmic /output:"%userdomain%-%computername%-share.txt" share list full /format:table
echo ...share done

wmic /output:"%userdomain%-%computername%-netuse.txt" netuse list brief
/format:table
echo ...netuse done

wmic /output:"%userdomain%-%computername%-ntdomain.txt" ntdomain list brief
/format:table
echo ...ntdomain done

wmic /output:"%userdomain%-%computername%-nic.txt" nic list full /format:table
echo ...nic done

wmic /output:"%userdomain%-%computername%-nicconfig.txt" nicconfig list full
/format:table
```

```
echo ...nicconfig done
```

This batch file writes information out into text files that are in "table" or space-delimited format. While .csv format might seem more useful for importing into other applications, the presence of many extra characters in the data (commas and others) normally used for delimiting, make this the easiest format to consistently import into Excel. The formatting options can be changed easily.

Some of the data collected may be redundant from one set to another. This batch file was designed for general purposes and is easily tailored to suit other purposes. Password hash dumping can also be added to such a batch file, assuming the needed permissions and program files are in place.

One of the most interesting results from this process was the following list of domain controllers.

| ClientSite | DcSite | Desc | DnsForest | DCAddress | DCName | Domain | Status |
|------------|--------|--------|-----------|---------------|-------------|--------|--------|
| PHASITE | PHAR01 | PHAR01 | pha.gov | \\10.10.1.55 | \\PHAR01DC2 | PHA01 | OK |
| PHASITE | PHAR02 | PHAR02 | pha.gov | \\10.20.1.4 | \\PHAR02DC1 | PHA02 | OK |
| PHASITE | PHAR03 | PHAR03 | pha.gov | \\10.30.1.4 | \\PHAR03DC1 | PHA03 | OK |
| PHASITE | PHAR04 | PHAR04 | pha.gov | \\10.40.1.4 | \\PHAR04DC1 | PHA04 | OK |
| PHASITE | PHAR05 | PHAR05 | pha.gov | \\10.50.1.55 | \\PHAR05DC2 | PHA05 | OK |
| PHASITE | PHAR06 | PHAR06 | pha.gov | \\10.60.1.55 | \\PHAR06DC2 | PHA06 | OK |
| PHASITE | PHAR07 | PHAR07 | pha.gov | \\10.70.1.4 | \\PHAR07DC1 | PHA07 | OK |
| PHASITE | PHAR08 | PHAR08 | pha.gov | \\10.80.1.4 | \\PHAR08DC1 | PHA08 | OK |
| PHASITE | PHAR09 | PHAR09 | pha.gov | \\10.90.1.55 | \\PHAR09DC2 | PHA09 | OK |
| PHASITE | PHAR10 | PHAR10 | pha.gov | \\10.100.1.55 | \\PHAR10DC2 | PHA10 | OK |
| PHASITE | PHAR11 | PHAR11 | pha.gov | \\10.110.1.55 | \\PHAR11DC2 | PHA11 | OK |
| PHASITE | PHAR12 | PHAR12 | pha.gov | \\10.120.1.4 | \\PHAR12DC1 | PHA12 | OK |
| PHASITE | PHAR13 | PHAR13 | pha.gov | \\10.130.1.55 | \\PHAR13DC2 | PHA13 | OK |
| PHASITE | PHAR14 | PHAR14 | pha.gov | \\10.140.1.55 | \\PHAR14DC2 | PHA14 | OK |
| PHASITE | PHAR15 | PHAR15 | pha.gov | \\10.150.1.55 | \\PHAR15DC2 | PHA15 | OK |

Domain controllers are of key interest to attackers because they contain the Active Directory list of password hashes for all users in

the domain. Once the Active Directory administrator account has been compromised, the entire domain has been compromised. This is the holy grail target for most attackers.

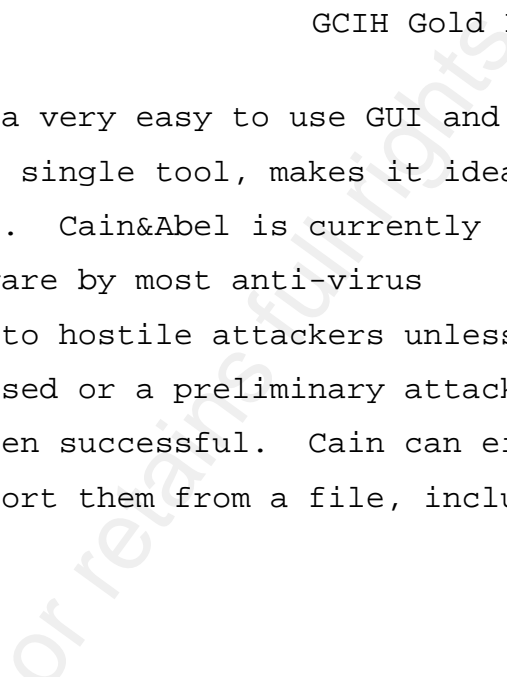
C - ATTACKERS: Password Cracking

Password cracking is generally described as the process of extracting logon password hashes and cracking them offline. Cracking often begins with a dictionary attack that checks to see if common words were used to create a password. Once a dictionary attack is exhausted, the next step is often a brute force attack that checks every possible combination of a specific character set. In either type of attack, the normal technique is to compute the hash from the current password guess, check to see if it is the same as the actual hash, then move on to the next guess.

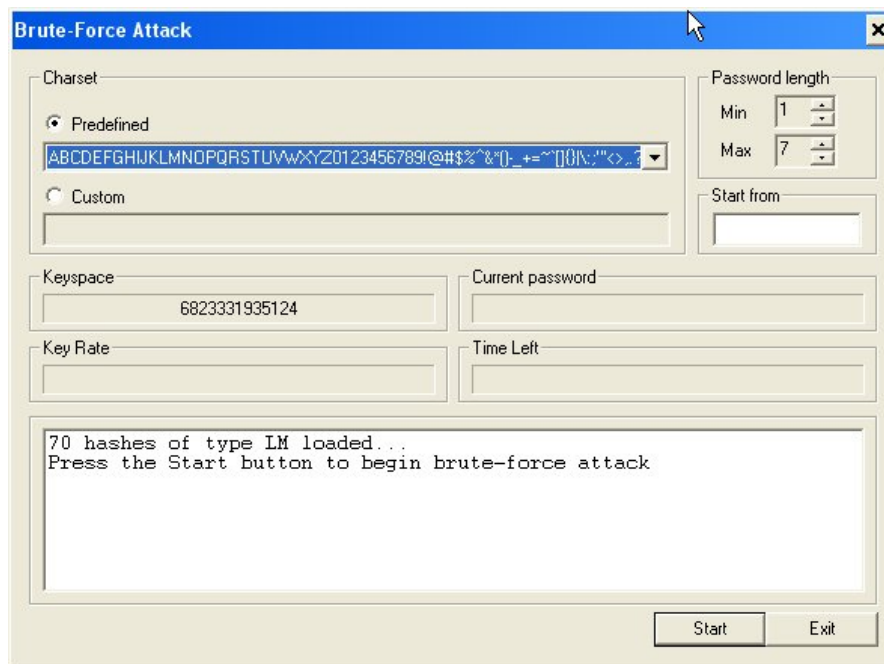
A particular weakness of windows systems is the storing of password hashes in the older "LM" format, which, prior to computing the hash, forces the password into all upper-case, then splits the hash into two seven character chunks, which makes it much easier to crack. The newer NT hash format does not inject these weaknesses and is harder to crack, but LM hashes are typically stored for backward compatibility. The first step in cracking hashes is extracting them from the system and that normally requires admin access privileges.

Cain&Abel is a multi-function tool that includes password hash extraction and cracking functions. It is not as powerful as some

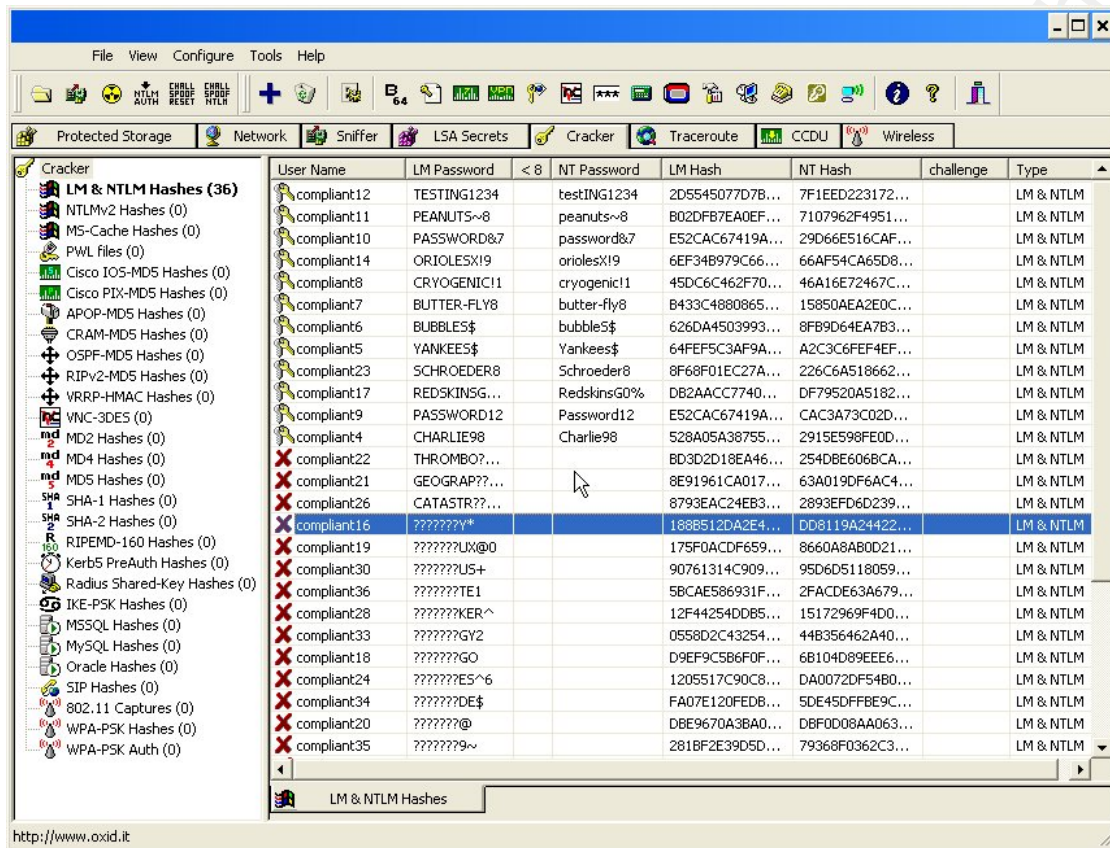
a very easy to use GUI and single tool, makes it ideal. Cain&Abel is currently are by most anti-virus to hostile attackers unless sed or a preliminary attack en successful. Cain can eort them from a file, inclu



a very easy to use GUI and single tool, makes it ideal. Cain&Abel is currently are by most anti-virus to hostile attackers unless sed or a preliminary attack en successful. Cain can eort them from a file, inclu



[brute force attack password cracking in Cain]



[passwords being cracked in Cain]

John the Ripper is a powerful password cracking tool available for both windows and linux. In its default mode it uses a hybrid of what it considers to be "best case mix" of dictionary and brute force attacks. This makes it very easy to launch john and come back hours or days later to view the cracked passwords. John also has powerful options that include the ability to calculate complex hybrid variations of dictionary files. This feature can be used to feed input into other cracking tools (such as wep crackers for wireless).

Rainbow crack tables are pre-computed tables of password hashes that greatly speed up the cracking process.¹⁸ In a normal brute force attack, the cracker program computes candidate hashes to compare against the real hash to determine success. This takes time. The rainbow table calculates all the possible hashes for a given character set ahead of time, creating very large tables, and then uses sophisticated lookup techniques to speed up access to them and allow quick confirmation. Some rainbow tables can crack alphanumeric password hashes in a matter of seconds. Even when you include special characters, LM format hashes can be cracked in a matter of hours.

Password cracking can be a powerful penetration weapon when you consider it as a stepping stone in a larger framework. A common administrative password is often used by different admin staff across many different systems for routine tasks. Sometimes, the hashes for these logins are left behind on workstations tended to by the admins, without their awareness. By penetrating a low priority workstation that has no valuable information on it, the attacker may be able to retrieve and crack a "maintenance" level admin password that gives him legitimate access to many other systems, including some that may contain password hashes at higher levels, even domain admin credentials.

D - Scenario (Entrench and Crack)

As the password hashes were collected, they were put into John the Ripper to see what would come out in a first pass of only ten to

fifteen minutes. Any NTLM hashes or LM hashes that took longer to crack could be split off to be handled separately.

```

C:\Program Files\john\john1701\run>john-mmxc compliant.txt
Loaded 72 password hashes with no different salts (NT LM DES [64/64 BS MMX])
SCHROED (compliant23:1)
BUBBLES (compliant6:1)
CATASTR (compliant26:1)
CRYOGEN (compliant8:1)
GEOGRAP (compliant21:1)
PASSWOR (compliant10:1)
PASSWOR (compliant9:1)
THROMBO (compliant22:1)
PEANUTS (compliant11:1)
REDSKIN (compliant17:1)
CHARLIE (compliant4:1)
TESTING (compliant12:1)
BUTTER- (compliant7:1)
GO (compliant18:2)
YANKEES (compliant5:1)
0 (compliant3:2)
4 (compliant13:2)
45 (compliant31:2)
90 (compliant29:2)
98 (compliant4:2)
1234 (compliant12:2)
ORIOLES (compliant14:1)

```

[in a matter of seconds, John's hybrid attack began producing cracked passwords - note that all of these passwords were compliant with the defenders' password complexity policy]

```

Command Prompt - john-mmxx compliant.txt
SCHROED (compliant23:1)
BUBBLES (compliant6:1)
CATASTR (compliant26:1)
CRYOGEN (compliant8:1)
GEOGRAP (compliant21:1)
PASSWOR (compliant10:1)
PASSWOR (compliant9:1)
THROMBO (compliant22:1)
PEANUTS (compliant11:1)
REDSKIN (compliant17:1)
CHARLIE (compliant4:1)
TESTING (compliant12:1)
BUTTER- (compliant7:1)
GO (compliant18:2)
YANKEES (compliant5:1)
0 (compliant3:2)
4 (compliant13:2)
45 (compliant31:2)
90 (compliant29:2)
98 (compliant4:2)
1234 (compliant12:2)
ORIOLES (compliant14:1)
$ (compliant5:2)
$ (compliant6:2)
% (compliant1:2)
* (compliant2:2)
@ (compliant20:2)
SECRET9 (compliant35:1)
PURPLE9 (compliant3:1)
ORANGE8 (compliant2:1)
FLY8 (compliant7:2)
6SKYWAL (compliant28:1)
D12 (compliant9:2)
GY2 (compliant33:2)
TE1 (compliant36:2)
ER8 (compliant23:2)
-4 (compliant25:2)
Y* (compliant16:2)
9~ (compliant35:2)
~8 (compliant11:2)
SG0% (compliant17:2)

```

[John continues cracking - note that the cracked hashes are handled in two different 7 byte parts - each password has a part one and a part two because of the LM format - you can see how easy it is to crack the often small "part 2"]

```

Command Prompt - john-mm compliant.txt
CHARLIE (compliant4:1)
TESTING (compliant12:1)
BUTTER- (compliant7:1)
GO (compliant18:2)
YANKEES (compliant5:1)
0 (compliant3:2)
4 (compliant13:2)
45 (compliant31:2)
90 (compliant29:2)
98 (compliant4:2)
1234 (compliant12:2)
ORIOLES (compliant14:1)
$ (compliant5:2)
$ (compliant6:2)
% (compliant1:2)
* (compliant2:2)
@ (compliant20:2)
SECRET9 (compliant35:1)
PURPLE9 (compliant3:1)
ORANGE8 (compliant2:1)
FLY8 (compliant7:2)
6SKYWAL (compliant28:1)
D12 (compliant9:2)
GY2 (compliant33:2)
TE1 (compliant36:2)
ER8 (compliant23:2)
-4 (compliant25:2)
Y* (compliant16:2)
9~ (compliant35:2)
~8 (compliant11:2)
SG0% (compliant17:2)
OW3R (compliant32:2)
CRYOLOG (compliant16:1)
DE$ (compliant34:2)
D&7 (compliant10:2)
X!9 (compliant14:2)
US+ (compliant30:2)
LUCYBIT (compliant24:1)
IC!1 (compliant8:2)
OPHEX9 (compliant26:2)
guesses: 50 time: 0:00:09:06 (3) c/s: 56080K trying: DEXWNNI - DEXWN-F

```

[results from John at the 9 minute mark]

There were no NTLM only hashes found in the collection and almost 70% of the LM hashes cracked in under 10 minutes. This was expected. The easily cracked LM passwords were dumped into a database and sorted and analyzed statistically in the hopes that some intelligent guessing might be gained or modifications to the dictionary lists suggested. The remaining hashes were dumped back into cracking mode using a rainbow table and yielded 100% of the remaining passwords in about three hours.

As the first round of cracking was completed, the attackers found that they had many passwords for individual accounts on workstations. They also noticed that the defenders seemed to use a

common login for routine administrative tasks, since this password hash was left behind on most of the systems. This login had administrative rights but was probably not the domain admin account. They tried it out on several servers and were able to dump the password hashes stored there. Using their list of domain controllers retrieved by the batch file, they went after the domain admin accounts. In some cases, the "routine" admin account worked to give them access to a domain controller and they were able to dump the complete set of hashes for the entire domain. In other cases, they had to work their way patiently up a tier of "stepping stones" from workstation to server to server to domain controller, but the end result was the same: they gained control of the domain admin accounts and the domain controllers. With a valid domain admin account at their disposal, they could access every part of the network with little fear of being detected. This is a "game over" condition.

9 - Zero Day

A. DEFENDERS: SI-4 INTRUSION DETECTION TOOLS AND TECHNIQUES

[the italicized section below is a security control from NIST SP 800-53]

Control: The organization employs tools and techniques to monitor events on the information system, detect attacks, and provide identification of unauthorized use of the system.

Guidance: Intrusion detection and information system monitoring capability can be achieved through a variety of tools and techniques (e.g., intrusion detection systems, virus protection software, log monitoring software, network forensic analysis tools).

Control Enhancement 1: The organization networks individual intrusion detection tools into a system wide intrusion detection system using common protocols.

Control Enhancement 2: The organization employs automated tools to support near-real-time analysis of events in support of detecting system-level attacks.

Control Enhancement 3: The organization employs automated tools to integrate intrusion detection tools into access control and flow control mechanisms for rapid response to attacks by enabling reconfiguration of these mechanisms in support of attack isolation and elimination.

Control Enhancement 4: The information system monitors outbound communications for unusual or unauthorized activities indicating the presence of malware (e.g., malicious code, spyware, adware).

[the following is the PHA response to the security control described above]

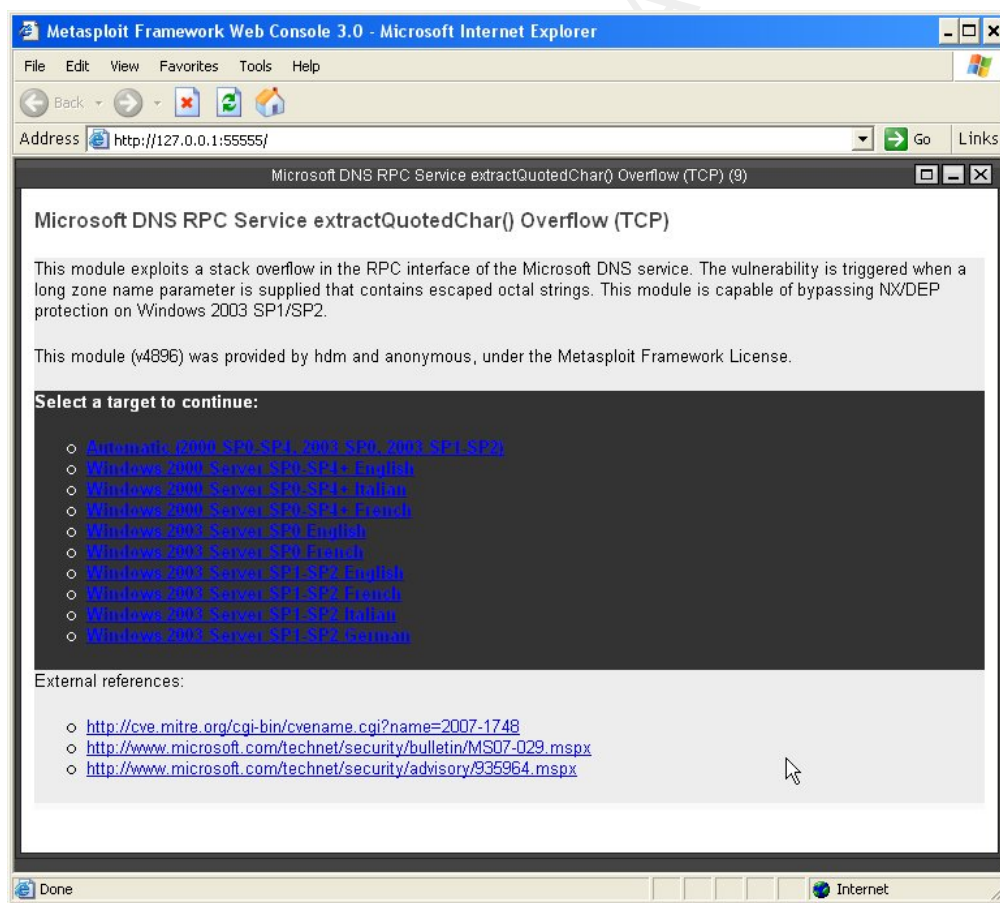
Implementation: The Office of Information Architecture as authorized by the Deputy Assistant Secretary for the Office of Information has directed the Office of Cyber Information and Security Compliance Assurance to be responsible for developing and deploying controls on an enterprise basis that protect PHA networks from penetration. This will include network intrusion detection devices to be deployed at each of the national network gateways to the internet. This will insulate the PHA internal network from cyber attacks.

B - ATTACKERS: Zero Day Exploits

A "zero-day" attack is an attack that targets a vulnerability for which there is no solution easily available. Once the vendor releases a patch, the zero-day exposure has ended. A recent example of a critical zero-day vulnerability was the Windows Animated Cursor Remote Execution Vulnerability that was patched by MS07-017¹⁹ (Microsoft Security Bulletin 925902). This was considered a critical hole because it could allow remote code of the attackers' choosing to be executed. A security research company called Determina notified Microsoft of the problem on December 20, 2006.²⁰ The vulnerability was publicly announced on March 28 2007.²¹ On April 2nd, Determina released a video demonstration of Metasploit using exploit code against Vista.²² Microsoft released the patch on April 3, 2007 ending at least six days of zero-day exposure. Exploit code that targeted this vulnerability was active in the wild for at least several days, if not several weeks before the patch was released.

Even after a patch is released, many organizations take several days to get around to updating systems with the patch.

Another recent example is the DNS RPC buffer overflow that was patched by MS07-029²³. This vulnerability offered remote code execution with SYSTEM access privileges against Microsoft DNS Server on both Win2000 and 2003. Exploit code was seen as early as April 7, 2007, Microsoft released a bulletin acknowledging the vulnerability on April 12, 2007 and the patch closed the hole on May 8, 2007, offering at least 31 days of zero-day opportunity to attackers who had exploit code. Metasploit had exploit code for this vulnerability included before the patch was released.



[Metasploit exploit for DNS RPC]

Earlier in the year, ImmunitySec (maker of Canvas) released an exploit for MS07-004²⁴ within hours of Microsoft's patch release. The exploit used a VML flaw in IE 7.0 to take over full control of the target system.²⁵ Over the past year or so, this sequence of events has become commonplace. A vulnerability is announced and nearly simultaneously we hear that there is active exploit code. Then we have to wait until a patch or workaround is released. Even when the patch or workaround becomes available, it takes time to deploy. This gives the attackers more than a few major zero-day vulnerabilities available each year if they are patient enough to wait a few weeks or a few months until the next one surfaces.

Gunter Ollmann, Director of Security Strategy at IBM Internet Security Systems has said, "all my consultants have access to over 100 0-days as a matter of course"²⁶. He continues, "For those people who say that the 0-day threat is fictional, or that their security system can prevent and contain any such outbreak, my response is 'dream on'" and adds, "Responses to successful 0-day penetration tests should be seen as live practices for disaster recover processes". eEyeDigital Security maintains a zero-day tracker web page that includes both active zero-day vulnerabilities and a history of older ones that have been fixed.

<http://research.eeye.com/html/alerts/zeroday/>

D. Scenario (Zero Day Attacks)

Both the attackers at the perimeter and the ones who had already penetrated the network through other means were waiting patiently for a zero-day vulnerability to become available. The perimeter attack team was holding just outside the network perimeter, but had collected more information about the perimeter gateway systems than the defenders would like. The other three attack teams were already inside the perimeter and had been working hard but quietly to further entrench their position without alerting any of the defenders. When the day finally arrived that the lab announced they had live zero-day exploit code, all four attack teams sprang into action. The exploit was plugged into various framework tools and launched. In a matter of minutes, the attackers found themselves at command prompts on dozens of compromised systems. They immediately began uploading the tools for entrenching their positions and launching more attacks against other systems. They could now attack almost with impunity, knowing that there were no defenses against this attack, including detection by IDS and AV software.

Within a few hours, over a hundred systems had been taken and over the next few days the count soared into the thousands. The preliminary zero-day window stayed open for over a week before the vendor made an announcement about it and it was another two and a half weeks before a patch was released. All told, the attack teams had been given free license to pillage the defenders network for over three weeks and many thousands of systems had been compromised. The attackers now owned most of the critical infrastructure of the

network and were confident that they had enough valid credentials to control all of it if they liked.

10 - Distributed Denial of Service Attack

A - ATTACKERS: DDOS

A standard Denial Of Service (DOS) attack denies access to some computing function of a system, or even access to the entire system, often by flooding a resource channel. When DOS attacks are launched from a single system and use normal networking protocols, the attack can usually be identified, traced back to the source and blocked or other action taken. In order to make it more difficult to take action against such attacks, Distributed DOS attack tools were designed. DDOS attacks often use the same techniques as DOS attacks but send them from many different sources. The sources are often controlled by a second layer of systems that relay commands but do not participate in the attack. The command nodes issue instructions to the actual attacking systems to vary their attacks both in type of attack and in timing. Using this system, the attack can achieve an effect of an entire spectrum of DOS attacks that is constantly shifting and changing. A node can be instructed to perform a smurf attack for a few minutes, then go silent for a few minutes, then resume with a syn flood for a few minutes, then go silent for a few minutes, then resume with yet another unique type of DOS attack (ICMP floods, UDP floods, DNS reflection attacks...), then go silent, and then repeat the process ad infinitum. Using ad-hoc networks of thousands of compromised systems that are called bot-nets, to launch a distributed attack, makes it unlikely that it will be easy to quench the attack or even filter it effectively.

Recently, as defenses have been developed to attack the command and control elements of bot-nets, they are evolving to use peer to

peer structures as a counter²⁷. They are starting to use encrypted communication channels and polymorphic stealth techniques that make them harder to eliminate.²⁸ A recent botnet/worm, called Nugache, has shown encrypted communications over an ad-hoc peer to peer network.²⁹

A recent paper discussed weaknesses in current botnet design (Nugache, Slapper, Sinit, and Phatbot) and presented a new design for an advanced hybrid peer to peer botnet. This design uses individualized encryption and ports, making it more difficult to detect through network flow analysis. It uses public key encryption for command authentication to prevent hijacking. Shifting command and report traffic across a network of many sensors makes it difficult to either intercept or block. Each bot contains a peer list for communication, but the list is kept short and never shared, minimizing the damage to the larger network if a bot is discovered and analyzed.

B - Scenario (DDOS)

The purpose of the attackers DDOS attack was to both disrupt normal network traffic and to demonstrate their control of the network to the extent that the defenders would be forced to shut the network down. As the attackers continued their penetration of the network, they had deliberately targeted key infrastructure components, including: email servers, IDS sensors, database servers, routers and switches, and of course the domain controllers. With many of these systems compromised, it would be easy to actually disable the network from functioning, but the goal was to actually

induce the defenders to turn it entirely off themselves, out of mistrust of both their data and their ability to control their own systems.

One of the key elements of the disruption plan was to edit medical data in the patient records database and to do it over a time-frame that meant backup tapes were also contaminated. Since the defenders were using a "father/grandfather" backup rotation system scheduled over a monthly time-frame, the contamination had to take place over a period of greater than a month. This effort had begun early on in the penetration, and was accomplished with maximum effort given to stealth, to prevent it from discovered too early. Once the full blown attack was launched, the compromised data was deliberately revealed to the defenders to sow mistrust in all of the patient data, including the backups and even paper records that had been recently printed from the data.

The plan of disruption was scaled to begin slowly a few weeks before the final attack and escalate over that period until a grand climax was reached. Small disruptions that were not very consequential were initiated sporadically. Only systems that were considered inconsequential to the rest of the plan were used, with the consideration that if they became suspect to the defenders, they might be taken off-line and of no further use to the attackers. One of the attackers' objectives at this point was to exhaust the defenders before the large final attack was launched. The small attacks were not designed to cause any major disruption, but simply to be constant annoyances that required attention and kept the

support staff busy and moving from one small problem to the next and always falling behind on their normal task schedule.

In addition to compromising systems and preparing to launch the DDOS attack, the attackers also took some actions designed to destroy the defenders' will and ability to function. From the initial reconnaissance, they knew the defenders' had two highly skilled rapid response and forensics teams that could be mobilized on short notice to fly into a site needing their special skills. Several days before the large attack was scheduled, a deep intrusion was deliberately revealed by the attackers in locations that were not geographically close to the facilities that were the real final targets. This tactic was designed to lure the defenders' two highly skilled incident response teams out of position.

Communications systems were disrupted in an ever increasing crescendo with the rest of the incidents. Email server backups were deleted whenever possible, and then email accounts were tampered with, email records were deleted on a random basis and some entire accounts were deleted. Internet gateways and proxy servers were likewise tampered with. Services were turned off and other services not needed were turned on. At first this did not cause any serious disruption, but as time went on, the tampering became more serious, with services being deleted and registry files corrupted, requiring more and more time involved with repair and restore operations and slower response end users attempts to use normal communications channels.

Key members of the defense teams and key decision makers had been singled out of the basic recon database and further focused collection had been done to identify their home addresses and phone numbers and as much information as possible on members of their families. This data was used to harass and threaten their families.

The final DDOS blitz was launched on a timetable according to the attackers' plans for their biological weapons attack. They wanted the network to be shut down on the same day as casualties began to stream into the hospitals. Since some DOS attacks can have their sources spoofed, it is impossible to know how many sources were actually involved in the attack, but it is clear that there were many hundreds and probably several thousand systems included in the internal botnet.

At the same time as the DDOS attack was launched, the attackers began changing administrative passwords on some of the infrastructure elements, particularly the routers. Unable to even attempt to filter traffic with these units, the defenders were forced to shut them down and begin to rebuild them. Eventually, a decision was made to shut down the entire network and start the process of rebuilding every system from scratch.

11 - Aftermath and Lessons Learned

A. DEFENDERS: Effectiveness

RA-3 RISK ASSESSMENT

The hospital network defenders actually had a good threat analysis completed that carefully considered real cyber threats, but then failed to use it in designing their security plan or in the controls that were actually implemented. Risk assessment is supposed to consider the attackers viewpoint and weigh their possible gain against their cost in order to create a determination of likelihood. It is an extremely "upstream" process that predicates and determines the general direction of all the other controls. If it is done incorrectly, it can influence every other element of the security plan. In this case, this upstream failure doomed all their other efforts by failing to realize the true nature of the threats they faced.

PL-2 SECURITY PLAN

The security plan appears to have been created in "boilerplate" fashion, copied from a template for all sites. This is okay as long as those parts are covering "common controls" that are specified by a central body, but not for any site specific parts. And if the common controls are done poorly, they will of course be done poorly for the entire enterprise. In this case, the security plans were in a shambles, representing paperwork only and full of errors even at that level. Most of the security plans showed little or no awareness of cyber attacks as a threat.

AC-11/12 SESSION LIMITS/TERMINATION

Session limiting controls were in place, but had little effect on limiting exposure because the timeouts were too long (at 15 minutes) and the controls were not supported by other strong controls such as physical security and security awareness. This problem was exacerbated by the fact that public hospitals have computers in non-restricted spaces. Session limits and security awareness need to be raised to more stringent levels in areas with public accessibility, and/or some other security measures, such as segmentation of the network and firewalls should be used. Proximity cards could be used to force logoffs when a user leaves a system.

AC-18 WIRELESS RESTRICTIONS

Wireless controls were in place, but had little effect in slowing down the penetration of the attackers because the technical level of the controls was not as advanced as the technical level of the attackers. Many of the standard wireless security measures (such as SSID cloaking and MAC address filtering) are trivial to defeat when the attackers have the correct knowledge and tools. WEP encryption is not safe in any configuration and can now be cracked in a matter of minutes, instead of hours. WPA encryption in conjunction with enterprise level authentication was needed.

IA-2 USER IDENTIFICATION AND AUTHORIZATION

Password strength settings were clearly defined in the security control, and well enforced by policy settings on the windows systems,

but they were not strong enough to prevent them from being cracked quickly and easily because of the LM hash format. The defenders knew about this weakness but had not taken action to update the plan, perhaps lulled to sleep by the false sense of security behind their strong perimeter defense.

IA-3 DEVICE AUTHENTICATION

This control was missing entirely and allowed rogue equipment to be plugged into live network ports and access the wired network with no intervention required.

PE-3 PHYSICAL ACCESS CONTROL

Physical protections were nearly 100% aimed at the main "computer room" and systems in publicly accessible areas were ignored. If the publicly accessible systems were handled differently, segmented from the rest of the network and treated with great concern, this might be okay. They were not.

SI-2 FLAW REMEDIATION

System updates were being done by an automated centralized tool, but not with a fast enough turnaround and many systems were apparently being missed by the tool and not tracked adequately. Even had they been done as fast as possible, with zero-day vulnerabilities becoming almost normal, this defensive component was becoming futile against professional attacks.

SI-3 MALICIOUS CODE

Anti-Virus software was deployed well and used properly, but easily bypassed when the attackers used customized malware without known signatures.

SI-4 INTRUSION DETECTION

Intrusion detection was being done, but not very well. With all the other weaknesses in the defensive scheme, this element became critical for detecting the attackers' presence inside the perimeter after successful penetrations. This was not well understood and the concept of IDS was poorly implemented and under utilized.

SANS teaches a PICERL process for incident handling that includes the following:³⁰

Prepare

Identify

Contain

Eradicate

Recover

Lessons Learned

When the "Prepare" phase of this process is done poorly, the rest of the process suffers and it becomes more difficult to detect and

respond to an incident. When the "Identify" phase fails, the rest of the process becomes irrelevant until the incident is actually identified. This can be catastrophic.

B. DEFENDERS: Results

Federal law specifies that each government agency must follow the Certification and Accreditation process. In accordance with this, the defenders had spent much time and energy on developing an all-encompassing set of policies that governed their network security. Some of these policies were clearly defined and some were not. Some of them were correct solutions for security issues and some were not. Most of the time, whether the policy was clear and correct or not, they were not creating effective defenses against attacks. The end result was that the PHA spent millions of dollars and employed hundreds of "security specialists" to produce policy and procedure based paperwork, and to perform security inspections that mostly focused on making sure policy and paperwork were in place and failed miserably to actually remedy the real security weaknesses.

In the end, the defenders had been forced to completely shut down their entire network of computer systems at the worst possible moment and it would take months to recover to a fully operational state. The cost of all this was immeasurable. The highest cost of all of course was in human lives lost by the hospitals that might have been saved had the hospitals been able to respond to the crisis in anything resembling a normal fashion. It is difficult to estimate

how many lives were cost simply by the magnification effect of the cyber attack alone (ignoring any casualties that would have been sustained by the bio-weapons attack with no cyber attack), but numbers were suggested in the range from many hundreds to many thousands. The stunning success of the attacks created havoc and insecurity that had long term effects on the stock markets and overall economy. There was an overwhelming political response and general uproar as everybody tried to assign blame and attach their own agendas to the flood of public opinion. A military mobilization and strikes against various related targets followed.³¹ When the dust of the immediate crisis began to settle, there was a massive wave of firings and resignations within the PHA and a full blown festival of lawsuits ensued. Victims and their families filed both criminal and civil suits against every figurehead in the government that could in any way be associated with the PHA. All of the PHA administrative staff and most of the senior network management and officials responsible for security were included. Class action lawsuits continued for years afterward and took many years to be completely settled.

C. DEFENDERS: Lessons Learned

While the policy and paperwork approach offers a complete and comprehensive methodology for performing network security, it is worthless without realistic application toward actual threat scenarios.

The risk assessment component can drive the entire process in either the right or wrong direction. In this case, the components of

threat analysis that include the viewpoint of the attackers (penetration testing, studying hacker techniques, attackers' psychology and motivation, war-gaming scenarios...) were entirely missing from the security planning process. A corrected threat analysis must then focus attention on the critical components of the defense to prevent successful attacks. In this case, most of the security controls discussed here are such critical points and must be reinforced as much as possible.

Even with the best possible defensive posture, the attackers might be able to penetrate a network and great attention must be given to intrusion detection (and extrusion detection³²) processes that are capable of detecting an attacker presence on the network AFTER a successful penetration. Response teams need to be trained in handling incursions in process, not just forensic analysis after the fact. A silent attacker presence scenario (where the attackers complete their penetration and decide to lie quietly in wait inside your network until the right moment arises to capitalize on their position or leverage it into an attack on another trusted network) may be even more dangerous than the outcome portrayed here. It has been suggested that current real cyber attackers are in fact doing just that.³³

D. ATTACKERS: Effectiveness

Perimeter Attack

This attack vector was 100% successful and while the decision to stop and wait for a zero-day vector was a good one, it may not have been needed. In all likelihood, considering the weakness of the defenders intrusion detection ability, direct attacks using old exploits that would have been seen immediately might have been successful too. This would have required using some other tactics, including disabling automated defenses such as HIPS and AV, but these techniques are possible. Detected intrusions would also require very quick pivot attacks to create entrenched positions on other systems before the defenders could react, but this also does not seem to be much of a problem. As long as the current situation with zero-day exploits stays in place, this will continue to be a viable attack vector.

Wireless Attack

This attack vector was wildly successful because of the weaknesses in the WEP encryption protocol. As defenders continue to learn and adjust to WPA and strong authentication, this vector will diminish. Bluetooth and other wireless mechanisms are now growing rapidly and offer many new vectors, just as 802.11 did in its early stages. The future success of wireless attacks will depend on migrating to new tactics and tools as the landscape shifts.

Bypass Attack

This vector was 100% successful and is likely to remain there for some time. Signature based defenses are simply too easy to defeat. The best alternative at the moment seems to be anomalous behavior

recognition and that too can be defeated by mimicking the appropriate behavior.

Walk-in Attack

While this vector was also 100% effective, it might be the easiest to defend against. However, the defenders failure to adequately adjust security defenses between private systems and systems in publicly accessible spaces left them wide open. The strongest advantage this attack has is the wide variety of attack options and social engineering opportunities available. Once physical security is completely tightened down, this vector can always shift toward extortion and bribery efforts.

E. ATTACKERS: Results

The penetration and disruption mission was a complete success. Everything the attackers tried worked well.

"What the ancients called a clever fighter is one who not only wins, but excels in winning with ease." Sun Tzu³⁴

F. ATTACKERS: Lessons Learned

Using the multi-pronged attack was great for training the field operatives, but involved far higher risk of discovery than was necessary. Future attacks will be designed to be more focused on

single weaknesses that have been identified by reconnaissance and perhaps even other missions (whether successful or failed).

Future attacks will be more stealthy in nature and will involve nearly impossible to detect extrusion techniques to export both enterprise data and command and control information for virtual botnets operating inside the defensive perimeter. Such a silent presence is likely to be far more valuable to a national security penetration team, unlike the terrorists who wanted to immediately use (and therefore eliminate) the inside position gained.

12 - Concluding Notes

The tools and tactics presented here are current, but not leading edge - real attackers are likely to be using more advanced tools and techniques. Many other existing vectors that could be considered as penetration pathways were not presented in this paper. Entire papers could be written about each of the security controls that have been presented here and each of the attack vectors used. A macro approach was necessary in order to create a collective version.

Consider this; the attackers are well trained, highly motivated professionals who are focused on a single purpose, use good knowledge management techniques and have a sophisticated understanding of attack methodology. They are going up against a team of defenders who are generally not well trained on security issues, are more motivated to please their boss or keep their jobs than to defend the network, have many tasks to perform besides security, often have office politics and bureaucracy inhibiting knowledge management and have little or no understanding of how they will be attacked. Ask yourself - if a professional national security cyber attack team has already penetrated YOUR network, would you know it?

13 - References

1 - Introduction

¹ Evers, Joris (2005, June 1). Panel paints grim picture of cybercrime battle. CNET news.com

² Rollins, John and Wilson, Clay. (2005, October 5) Terrorist Capabilities for Cyberattack: Overview and Policy Issues. CRS Report for Congress - order code RL33-123.

³ Magregor, Pat. (2001, October 3). Cyberterrorism: The Bloodless War? Powerpoint presentation

⁴ Sutherland, Blake. (2006, December 4). Enemy at the Gates! Your Computer Systems Are Under Attack. Radiology Today, Vol. 7 No. 24 P. 12.

2 - Preparation

⁵ Ross, Ron et al. (2006, December). Recommended Security Controls for Federal Information Systems, NIST Special Publication 800-53. National Institute of Standards and Technology, U.S. Dept of Commerce.

⁶ Ross, Ron et al. (2007, June). Guide for Assessing the Security Controls in Federal Information Systems, NIST Special Publication 800-53A, Third Public Draft. National Institute of Standards and Technology, U.S. Dept of Commerce.

⁷ Bowen, Hash & Wilson. (2006, October). Information Security Handbook: A Guide for Managers, NIST Special Publication 800-100. National Institute of Standards and Technology, U.S. Dept of Commerce.

⁸ Ibid., 88.

⁹ Hammond, Grant T. (2001). The Mind of War, John Boyd and American Security. Smithsonian Books.

¹⁰ Sun Tzu. (6th century BC). The Art of War. Retrieved from http://en.wikiquote.org/wiki/The_Art_of_War

¹¹ Hammond, Grant T. (2001). The Mind of War, John Boyd and American Security. Smithsonian Books.

¹² Schmitt, John F. (1989). Fleet Marine Forces Manual-1 Warfighting. U.S. Marine Corps.

3 - Using Google for Reconnaissance

Long, Johnny. (2005). Google Hacking for Penetration Testers. Rockland, MA. Syngress Publishing.

Long, Johnny. (2007). Google Hacking Database. Retrieved June 10, 2007 <http://johnny.ihackstuff.com/ghdb.php>

4 - Perimeter

¹³ Metasploit. The Metasploit Project. Downloaded from <http://www.metasploit.com>

¹⁴ Nmap. Insecure.org. Downloaded from <http://insecure.org/nmap>

¹⁵ Nessus. Tenable Network Security. Downloaded from <http://www.nessus.org>

5 - Wireless Network

Kershaw, Mike. Kismet. Downloaded from www.kismetwireless.org

Backtrack2. RemoteExploits. Downloaded from www.remoteexploits.org

Aircrack. Downloaded from <http://www.aircrack-ng.org>

Vladimirov, Gavrilenko, & Mikhailovsky. (2004). WI-FOO, The Secrets of Wireless Hacking. Addison Wesley.

Peikari, Cyrus and Fogie, Seth. (2003). Wireless, Maximum Security. SAMS.

Hurley, Thornton, Puchol, & Rogers. (2004). Wardriving, Drive, Detect, Defend. Syngress.

Gast, Matthew. (2005). 802.11 Wireless Networks: The Definitive Guide, Second Edition. O'Reilly.

6 - Bypass

¹⁶ Rutkowska, Joanna. (2006, March). Rootkits vs. Stealth by Design Malware. Powerpoint presentation from BlackHat Europe 2006, Amsterdam.

Heyman, Karen. (2007, April 23). New Attack Tricks Antivirus Software. Computer Magazine - IEEE Computer Society. Retrieved from http://www.computer.org/portal/cms_docs_computer/computer/homepage/May07/COM_018-020.pdf

Evers, Joris. (2006, October 13). The future of malware: Trojan horses. CNET Networks.

Skoudis, Ed. (2006, December 13). What Are Polymorphic Viruses? TechTarget. Retrieved from http://searchsecurity.techtarget.com/expert/KnowledgebaseAnswer/0,289625,sid14_gcil247142,00.html

Jackson, Don. (2007, March 21). Gozi Trojan. SecureWorks. Retrieved from <http://www.secureworks.com/research/threats/gozi>

Prahlad Fogla, Monirul Sharif, Roberto Perdisci, Oleg Kolesnikov and Wenke Lee. (2006, August). Polymorphic Blending Attacks. 15th Usenix Security Symposium. Vancouver, BC, Canada.

Secureworks. (2004, August). Hackers make the evolutionary leap - Download.Ject signals new wave of attack methods. Retrieved from <http://www.secureworks.com/research/newsletter/2004/08/>

Giani, Berk and Cybenko. (2006) Data Exfiltration and Covert Channels. Giani_SPIE2006.pdf. Thayer School of Engineering, Dartmouth College, Hanover, NH 03755 USA

Koot, Matthijs and Smeets, Mark. (2006, February 5). Research Report: Covert Channels. University of Amsterdam

8 - Entrench

Ed skoudis on SANS - "Windows Command-Line Kung Fu with WMIC"
<http://isc.sans.org/diary.html?storyid=1229>

Microsoft. (2007). Using the Windows Management Instrumentation Command-line (WMIC) tool. Retrieved from <http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/wmic.msp>

Microsoft. (2007). Wmic. Retrieved from
<http://msdn2.microsoft.com/en-us/library/aa394531.aspx>

Johansson, Jesper & Riley, Steve (2005) *Protect Your Windows Network - from perimeter to data*. New Jersey: Addison-Wesley

¹⁷ Pwdumpx. Retrieved from <http://reedarvin.thearvins.com/tools.html>

¹⁸ Oechslin, Phillippe (2005) *Password Cracking: Rainbow Tables Explained* Retrieved from <https://www.isc2.org/cgi-bin/content.cgi?page=738>

9 - Zero Day

¹⁹ Microsoft. (2007). Microsoft Security Bulletin MS07-017. Retrieved from <http://www.microsoft.com/technet/security/Bulletin/MS07-017.mspx>

²⁰ Sotirov, Alexander. (2007). Windows Animated Cursor Stack Overflow Vulnerability. Determina Security Research. Retrieved from
<http://www.determina.com/security.research/vulnerabilities/ani-header.html>

²¹ Naraine, Ryan. (2007, March 30). Microsoft knew of Windows .ANI flaw since December 2006. retrieved from
<http://blogs.zdnet.com/security/?p=143>

²² Sotirov, Alexander. (2007). Exploiting Vista with ANI. Determina Security Research. retrieved from <http://determina.blogspot.com/2007/04/exploiting-vista-with-ani-html>

²³ Microsoft. (2007). Microsoft Security Bulletin MS07-029. Retrieved from <http://www.microsoft.com/technet/security/Bulletin/MS07-029.msp>

²⁴ Microsoft. (2007). Microsoft Security Bulletin MS07-004. Retrieved from <http://www.microsoft.com/technet/security/Bulletin/MS07-004.msp>

²⁵ Naraine, Ryan. (2007, January 11). Exploit Released for Critical PC Hijack Flaw. eWeek.com

²⁶ Ollman, Gunther. (2007). The 0-day Blues. Retrieved from <http://www.technicalinfo.net/opinions/opinion030.html>

10 - Distributed Denial of Service

²⁷ Nariane, Ryan. (2006, October 6). Is the Botnet Battle Already Lost? eWeek.com.

²⁸ Higgins, Kelly Jackson. (2007, January 7). Botnets Don Invisibility Cloaks. Dark Reading.

²⁹ Lemos, Robert. (2006, May 02). Bot Software looks to improve peerage. Retrieved from www.securityfocus.com/news/11390

Dittrich, Dave. (2007, June 12). Distributed Denial of Service (DDoS) Attacks/tools. Retrieved from <http://staff.washington.edu/dittrich/misc/ddos/>

Vaas, Lisa. (2007, April 16). Researchers: Botnets Getting Beefier. eWeek.com. Retrieved from <http://www.eweek.com/article2/0,1759,2114741,00.asp>

Nazario, Jose. (2007). Botnet Tracking: Nazario, Jose "Botnet Tracking. Black Hat DC 2007.

11 - Aftermath and Lessons Learned

³⁰ Skoudis, Ed and others. (2005). SANS courseware, SEC 504 Hacker Techniques, Exploits & Incident Handling. SANS Institute.

³¹ Messmer, Ellen. (2007, February 9). RSA - US cyber counterattack: Bomb one way or another. Retrieved from <http://www.networkworld.com/news/2007/020807-rsa-cyber-attacks.html>

³² Bejtlich, Richard. (2005). Extrusion Detection. Addison Wesley.

³³ Rogin, Josh. (2007, Feb 13). Cyber officials: Chinese hackers attack 'anything and everything'. Federal Computer Week.

³⁴ Tzu. (6th century BC). The Art of War. Retrieved from http://en.wikiquote.org/wiki/The_Art_of_War

