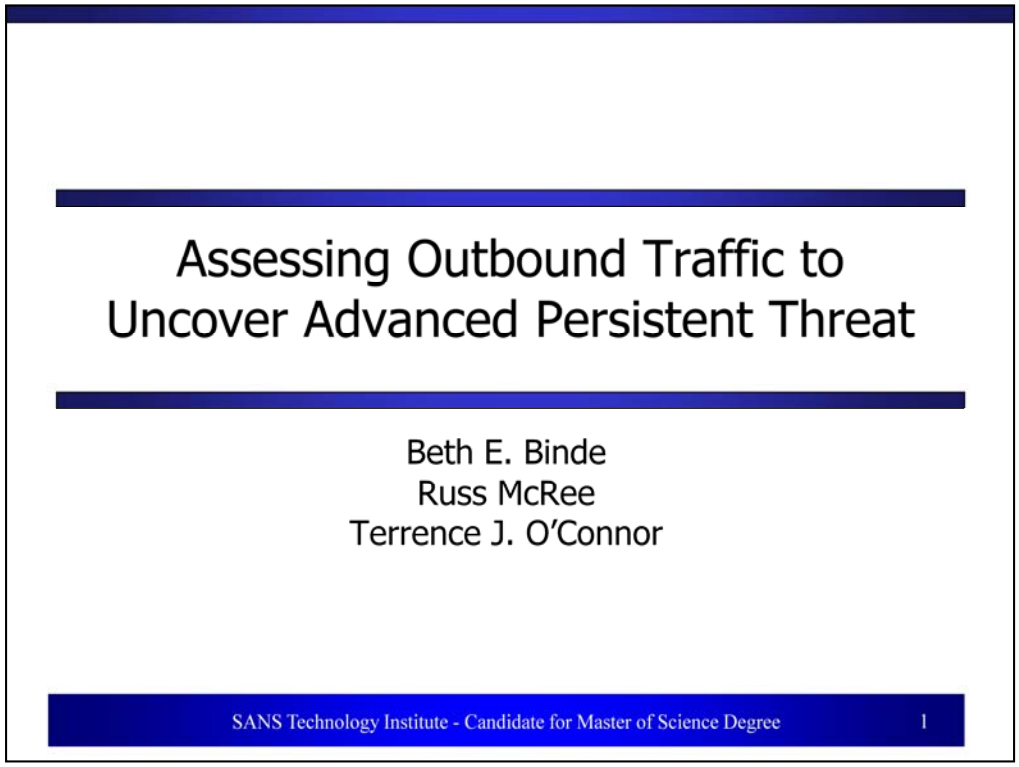


Advanced Persistent Threat (APT) exhibits discernible attributes or patterns that can be monitored by readily available, open source tools. Tools such as OSSEC, Snort, Splunk, Squil, and Squert may allow early detection of APT behavior. The assumption is that attackers are regularly attempting to compromise enterprises from basic service abuse to concerted, stealthy attempts to exfiltration critical and high value data. However, it is vital to practice heightened operational awareness around critical data and assets, for example, card holder data, source code, and trade secrets. Segment and wrap critical data within the deeper protection of well monitored infrastructure. Small, incremental efforts, targeted at protecting high value data value (typically through smaller and protected network segments), provide far greater gains than broader, less focused efforts on lower value targets. In a similar vein, layered defensive tactics (multiple layers and means of defense) can prevent security breaches and, in addition, buy an organization time to detect and respond to an attack, reducing the consequences of a breach.

Even the best monitoring mindset and methodology may not guarantee discovery of the actual APT attack code. Instead, the power of more comprehensive analysis and correlation can discover behavior indicative of APT-related attacks and data exfiltration. APT examples provided herein include Operation Aurora, the recent RSA. Additional opportunity for discussion include the LizaMoon attacks (SQL injection as an entry vector) as well as analysis specific to how FastFlux traffic might be indicative of deeper malfeasance. These additional considerations are important as they may serve as an APT entry point, or indicate its presence. The defining premise of this paper will be to:

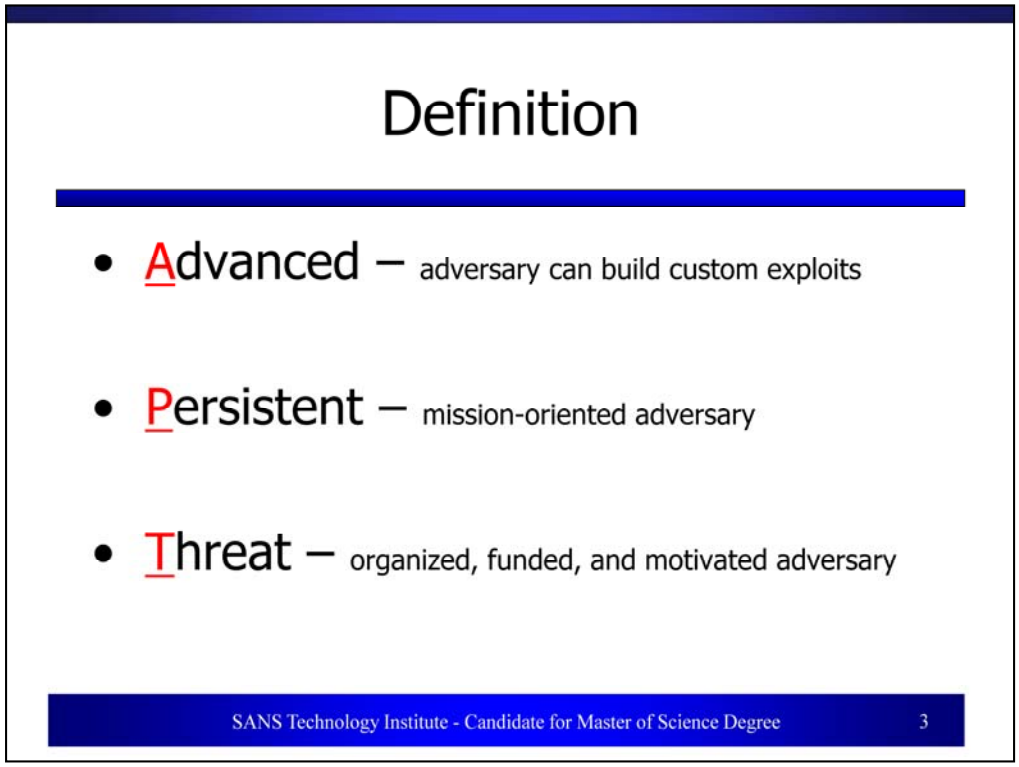
define and describe advanced persistent threat (APT)

propose technical approaches to mitigating the threat

include tools useful in the possible detection of APT

Objective

- Definition
- Operation Aurora
- RSA Breach
- Rules for APT
- Statistical and Correlation Methods
- Manual Approaches
- Automatic Prevention
- Summary



In 2006, the United States Air Force (USAF) analysts coined the term **advanced persistent threat** (APT) to facilitate discussion of intrusion activities with their uncleared civilian counterparts. Thus, the military teams could discuss the attack characteristics yet without revealing classified identities. [Bejtlich, 2007] Bejtlich explains the components of the terminology.

Advanced means the adversary is conversant with computer intrusion tools and techniques and is capable of developing custom exploits.

Persistent means the adversary intends to accomplish a mission. They receive directives and work towards specific goals.

Threat means the adversary is organized, funded and motivated.

Further, objectives may be political, economic (for example, the theft of intellectual property), technical or military (identification of weaknesses). [Bejtlich, 2010] *The Anatomy of an Advanced Persistent Threat* [Cutler, 2010] describes the typical APT strategy.

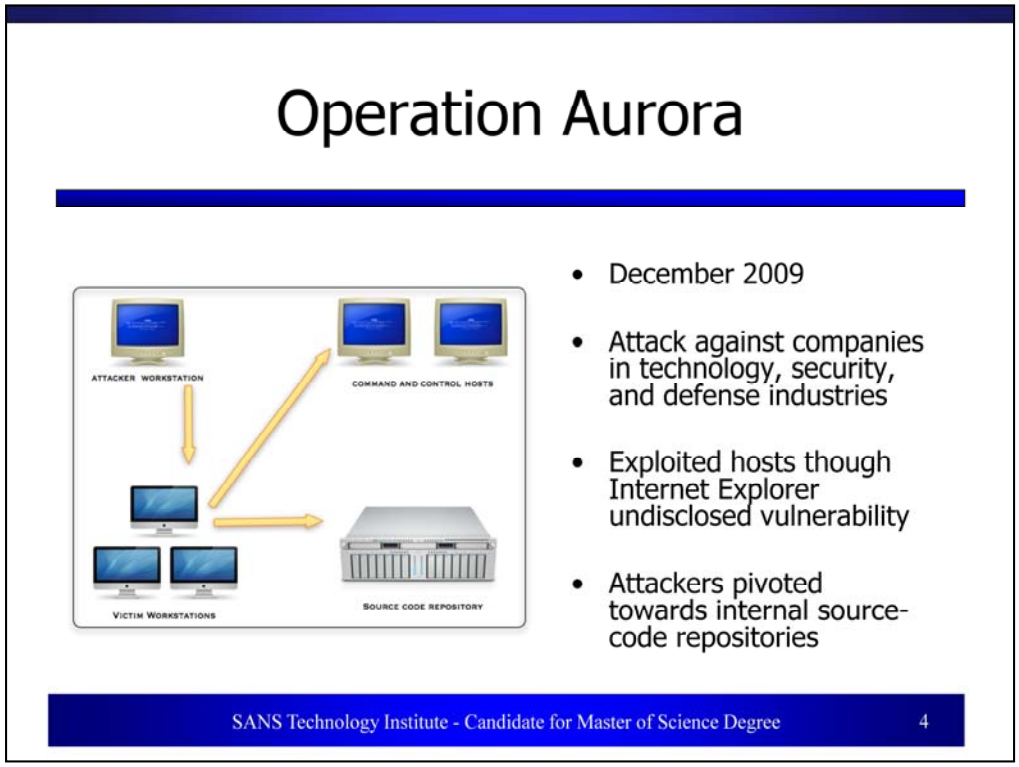
Attacker gains foothold on victim system via social engineering and malware

Attacker then opens a shell prompt on victim system to discover if system is mapped to a network drive

Victim system is connected to the network drive prompting attacker to initiate a port scan from victim system

Attacker will thereby identify available ports, running services on other systems, and identify network segments

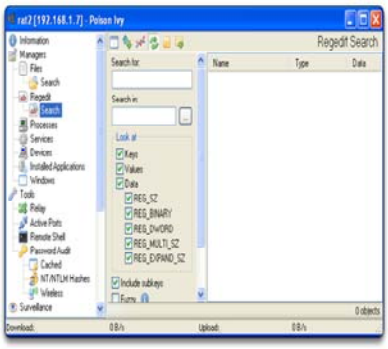
Network map now in hand attacker moves to targeting VIP victims with high value assets at their disposal



Operation Aurora demonstrates several of the key components of an advanced persistent threat (APT) based attack. [McAfee, 2010] This cyber-attack against several companies in the technology, security and defense industries started in mid-2009 and continued through December 2009. [Operation Aurora, 2011] Understanding that the intent of an APT is to gain access to targeted information and maintain a foothold in the environment for future use and control, Aurora is an excellent example to examine. [Daly, 2009] In the case of Aurora, attackers targeted the software-configuration management (SCM) systems that held proprietary information of Google, Adobe and other Fortune 100 companies over several months. The anatomy of the attack categorizes it as a classic APT attack.

RSA Breach

- Targeted attack against RSA network.
- Exploited hosts via an Adobe Flash undisclosed vulnerability embedded in a Microsoft Excel document
- Combined with targeted spear phishing campaign
- PI-RAT controlled hosts.



SANS Technology Institute - Candidate for Master of Science Degree 5

In March 2011, RSA acknowledged a successful attack against the RSA network. In an open letter addressed to customers, Art Coviella, the Executive Chairman, declared that the attack was APT. [Coviella, 2011] This attack further illustrates the key components of an APT attack and some of the methods that could be used to identify a similar attack.

The RSA attack began with a successful phishing campaign. This is a common technique for introducing malware into the target network. In this campaign, the attackers targeted two small groups of employees, delivering the phishing email over a two-day period and bypassing the installed email filters.

After successful exploitation via the infected spreadsheet, the victim workstations installed a remote access toolkit (RAT) known as Poison Ivy; also known as PI-RAT.

Rules - OSSEC

- OSSEC can monitor host based logs and activity to identify suspicious behavior
- Can detect PI-RAT (used in RSA Breach & Ghosnet attacks) by examining the contents of specific Windows Registry Keys and file installation paths.

** Alert 1305369454.50708: - ossec,rootcheck,
2011 May 14 06:37:34 (HIOMALVM02.7) 192.168.1.7->rootcheck
Rule: 514 (level 2) -> 'Windows application monitor event.'
Src IP: (none)
User: (none)
Application Found: Possible PoisonIvy-RAT. File: C:\Documents and Settings\All Users\Application Data\Microsoft\Network\Connections\Pbk\rasphone.pbk.

Rules – SNORT IDS

- Identify the **exploit** (PI-RAT shellcode)

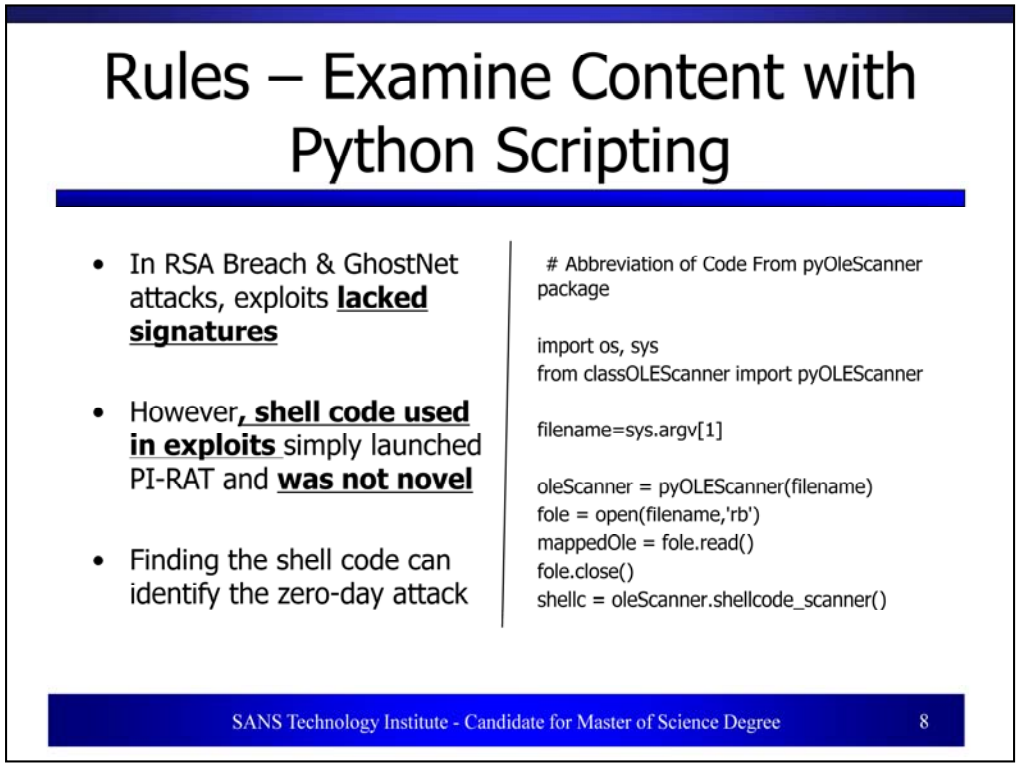
```
alert tcp any any -> $HOME_NET any (msg: "Poison
    IVY shellcode"; content: "|55 8B EC 8F C4 30 F0
    FF FF 60 33 C0 8D BD 84 F0|"; sid: 10000001;
    rev: 1;)
```

- Identify the **C&C channel** (PI-RAT C&C)

```
alert tcp $HOME_NET -> $EXTERNAL_NET 3460
    (msg: "Poison IVY Reverse Connection"; flow:
    from_client,established; content:"U|SB EC|P|B8
    02 00 00 00 81 C4 04 F0 FF FF|"; depth:15;
    sid:10000001;)
```

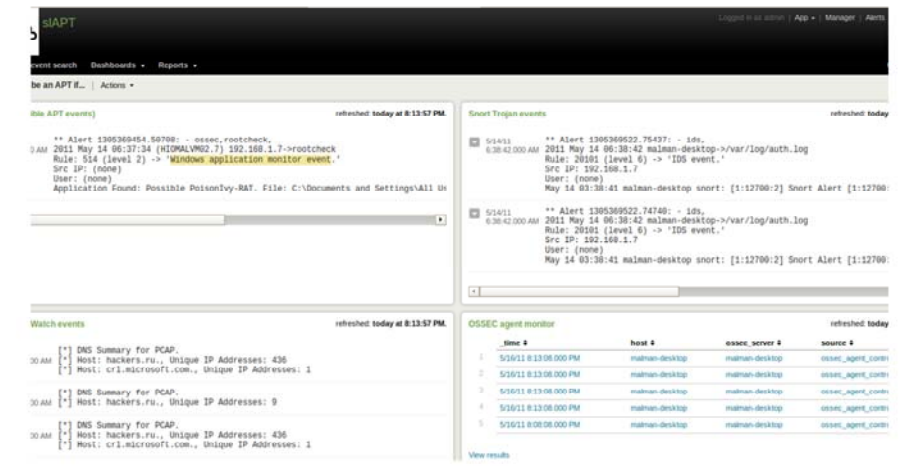
- Identify the toolkit **activity** (RAR file exfil)

```
alert tcp $HOME_NET any -> $EXTERNAL_NET any
    (msg:"ET POLICY RAR File Outbound"; flow:
    established; content:"|52 61 72 21|"; offset: 0;
    depth: 4; tag: session; classtype: not-
    suspicious; sid: 2001950; rev:3;)
```

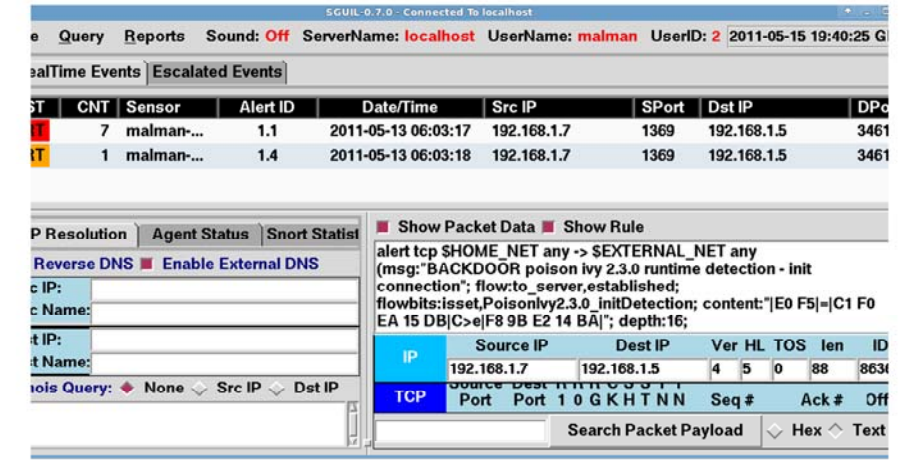


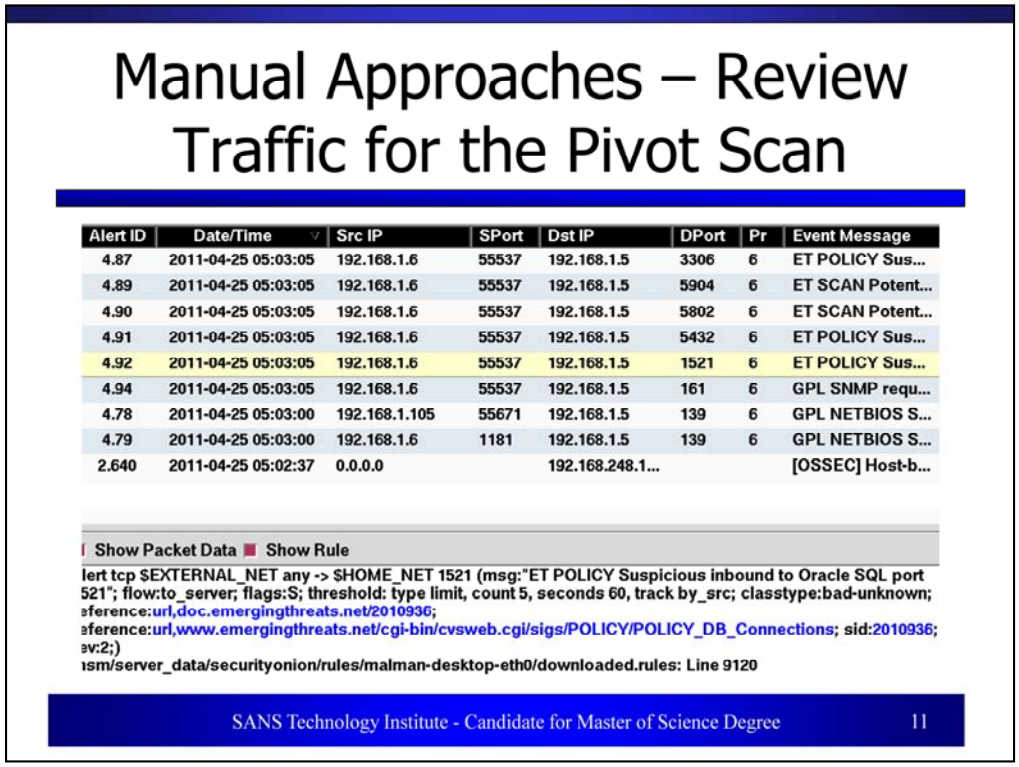
The pyOleScanner framework [Bonfa, 2011] can examine Microsoft ® Office documents further for suspicious content. Examples include known APIs (application programming interfaces), embedded structures, portable executable content, shellcode or XOR encrypted data. pyOLEScanner can be easily modified to scan the structures, identify suspicious content and prevent its delivery to the end user. [O'Connor, 2011]

Statistical Correlation: slAPT



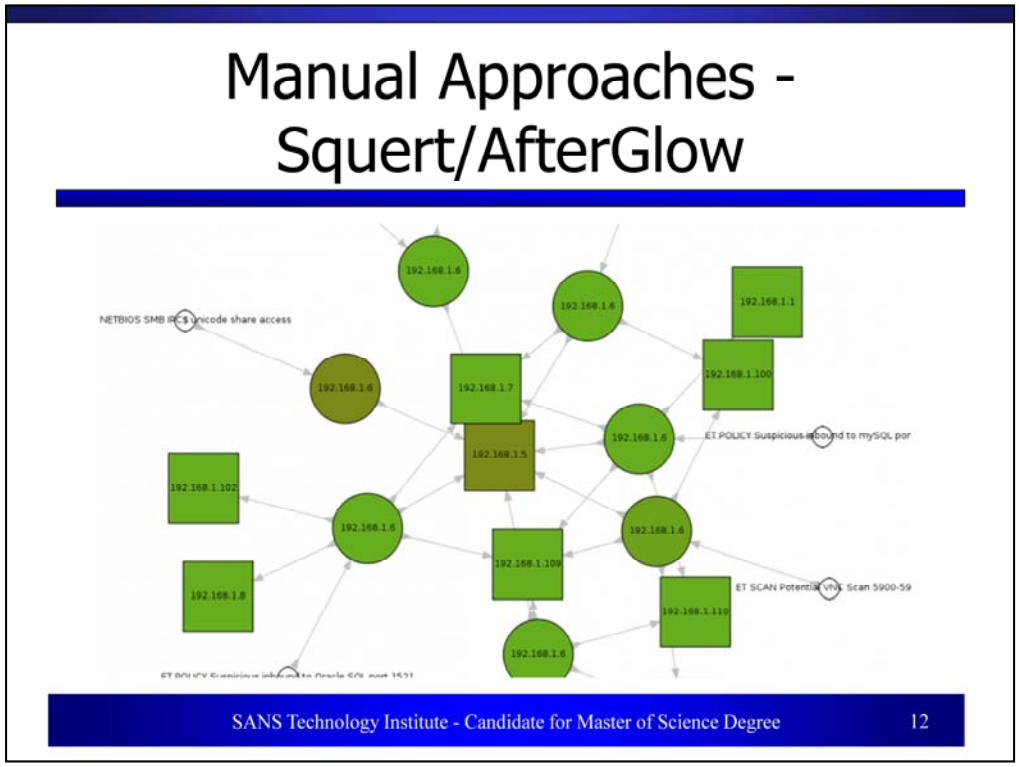
Statistical Correlation: Sguil





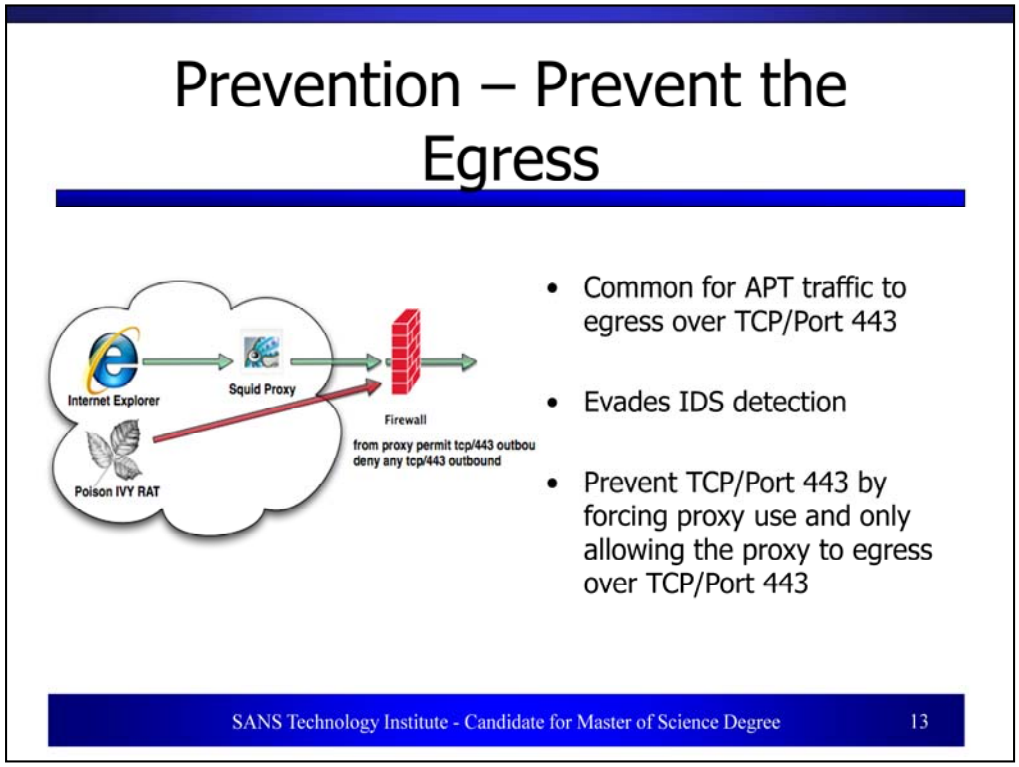
An attack such as Operation Aurora attack might include pivot scanning from an initially compromised host. While the initial host compromise might go undetected, what if network sensors spotted internal network scanning traffic that triggered an investigation in turn discovering a compromised host that otherwise may have gone unnoticed?

The above mentioned behavior should be identified assuming appropriate monitoring and analysis. A typical Sguil view of such port scanning traffic is noted in Figure 7. Note the Emerging Threats rules flagging it as a combination of scan and policy alerts. Consider that in tightly controlled networks (those given the utmost priority due to data value), port scanning from hosts other than those with explicit permission to do so should trigger immediate alerts and escalation.



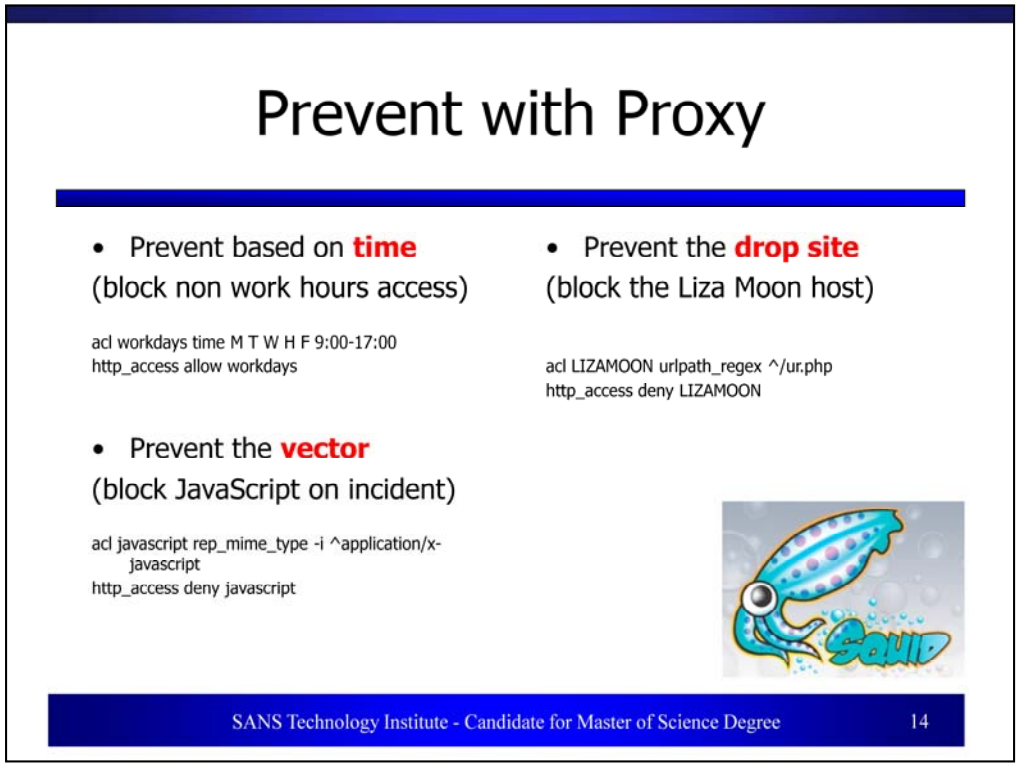
Many commercial security information and event managers (SIEM) offer additional functionality as it pertains to host that may be placed on a watch list, either manually or via a predefined rule set that automatically classifies certain behaviors and increases its scrutiny of the suspect host.

While the likes of Security Onion, as good as it is, doesn't provide the same level of functionality one might expect from a commercial product, it still offers certain feature inherent to those products. Many commercial vendors are now supplementing detection and alerting with visualization techniques. We contend that certain free and open source tools (FOSS) have been meeting the needs of security visualization practitioners for years. Security Onion includes Squert which in turn makes use of AfterGlow and the graphviz libraries to provide on demand visualizations of captured traffic. Again, making use of the premise of an attacker scanning from a beachhead host (pivoting), related scanning traffic from the pivot host presents itself in a tidy visualization. Note that, as seen in Figure 8, the victim pivot host 192.168.1.6 is probing its network neighbors for well known services that the attacker can then exploit if vulnerable.



In the fight against APT, one method to prevent the attacker from succeeding in the exfiltration of data is to integrate a proxy into he environment. Squid is one such example of a proxy. (Squid, 2011) Squid can support HTTP, HTTPS, and FTP protocols and help as an integral part of a layered defense. Before diving too much into the specific methods for blocking and prevention, consider how Squid couple with a finely tuned firewall can work to prevent the APT persistence.

Consider the following scenario. A user inside your perimeter has Internet Explorer running and is proxying HTTP/S requests through a Squid Proxy before egressing the network. The firewall is enabled to allow only the Squid Proxy to initiate TCP sessions outbound on TCP Port 443. Even if the user is exploited and a remote administration toolkit such as PI-RAT is installed – it will fail to egress the network on TCP Port 443 because only the Proxy is permitted to leave the network on TCP Port 443.



Squid uses a series of access control lists (ACLs) to permit or deny traffic. Lets examine how these ACLs can be useful in the prevention of APT. One interesting statistic about APT is that the attacks frequently occur between 10 pm – 4 am EST when there is limited monitoring of the network and limited use by the intended victims. [Mandiant Corporation, 2011]. If your organization can afford to do it, limiting users' to a period of time when the business is open can certainly assist in the prevention of APT.

acl workdays time M T W H F 9:00-17:00

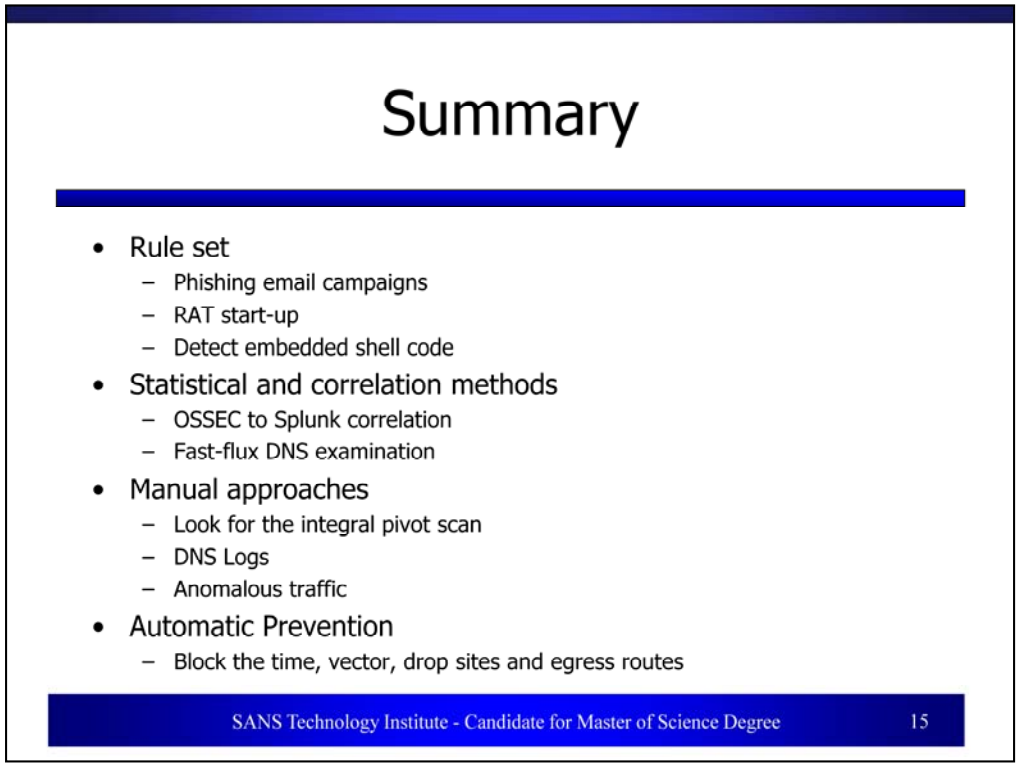
http_access allow workdays

While limiting users is certainly a good first step, opportunities will present themselves when a novel attack does enter your network. Squid ACLs can further assist by preventing the attack from succeeding against multiple victims. Consider the Aurora breach, where attackers exploited a use-after-free vulnerability in the JavaScript interpreter in Microsoft ® Internet Explorer ©. An administrator can immediately prevent JavaScript from functioning on vulnerable browsers in his/her organization by adding two lines of code of the Squid configuration as follows:

Acl javascript rep_mime_type -i ^application/x-javascript

http_access deny javascript

Additional rules for specific exploits can even be generated to allow for automatic blocking. In the case of the Liza Moon mass infection, infected sites contained an IFRAME to a PHP script ur.php. To safeguard users from the threat of LIZA MOON accidentally infection, an administrator can simply configure a regular expression to deny access to any ur.php pages from browsers using the Squid Proxy.



We have endeavored to provide four distinct approaches for consideration as possible countermeasures to the advance persistent threat, a pervasive, worrisome, and maliciously intentional attack against specifically targeted victims and data. These approaches have included well known signature based methodology, manual analytical practices, statistical tactics and correlation concepts, as well as automatic leak prevention.

While there is no silver bullet in the fight against concerted and targeted attacks, a holistic framework that includes varied methodology while embracing layered defensive tactics can prove fruitful. On a battlefield where innocent victims still readily fall prey to social engineering, and enterprises still fail to patch vulnerabilities in a timely manner, the advanced persistent threat will do just that: persist. Yet concepts as described above, implemented with free, open source tools or supported, commercial platforms, coupled with comprehensive and steady analysis, can help turn the tide. While the war may never be won, perhaps some battles can be.