Global Information Assurance Certification Paper

Orion Incident Response Live CD

GIAC (GCIH) Gold Certification

Author: John Jarocki, john.jarocki@gmail.com
Advisor: Rodney Caudle

Accepted: April 24th 2010

Abstract

Computer intrusion response often requires working in hostile environments. In an ideal situation, the defender would work on trusted systems, with trusted – even out-of-band – communications channels. This paper assumes a non-ideal situation that more likely matches the norm. In this environment, everything is suspect: servers might be compromised, clients might be hostile, and the network itself could be suspect. The proposed solution is a custom-built, persistent Live CD pre-installed with incident response and analysis tools on a platform that allows strong authentication and encrypted communication with other defenders in the line of fire.

Orion is a prototype Live CD-based system intended to provide a self-contained, trusted platform for incident response team members to use for analysis, communication, and collaboration. Orion is currently based on the BackTrack Linux distribution from Offensive Security. While BackTrack is focused on Penetration Testing, Orion is focused on incident response and defense. In security parlance, BackTrack is built for Red Team, while Orion is built for Blue Team.

1. Introduction

There are many frameworks for incident handling, including the Security Incident Handling Guide from the National Institute of Standards and Technology (NIST) (Scarefone, Grance, & Masone, 2008), (Mandia, Prosise, & Pepe, 2003), and the SANS six-step handling process (Skoudis, 2009). Carnegie-Mellon's Software Engineering Institute even provided a study of these and more along with a framework for creating an incident management process tuned for a specific organization (Albert, Dorofee, Killcrece, & Zajicek, 2004). Tools for incident handling and response also exist, but responder tool kits are often built by collecting important tools one at a time until the expert incident handler has a custom set. This makes it difficult to bring in less-experienced incident response team members, and it creates challenges for collaboration during the incident as well as consistent collection and storage of evidence. Some Incident Response environments do exist, but they primarily focus on the analysis process, and do not provide a communication and collaboration framework. Nor do they usually provide a workflow based environment.

In a small incident, experienced handlers will compensate for challenges in communication and collaboration. However, part-time team members or individuals who are new to the incident handling process will find it extremely difficult to operate in a restrictive, need-to-know environment where communication is limited to only trusted channels and where consistent collection and storage of information is critical to the success of the process. Even full-time, experienced responders will find Orion useful for helping to keep efforts focused and consistent as the incident drags on into the wee hours of the night.

The Orion Incident Response system was created to provide a trusted incident response platform that provides secure communication channels and collaboration tools in a consistent environment that can be used by all team members.

John Jarocki, john.jarocki@gmail.com

2. Inspiration

The idea for Orion started with the realization that incident response teams face a number of challenges that are not typically addressed, except in an ad-hoc fashion:

- Some incident responders are more experienced than others. Experienced members of the team often need to delegate tasks to junior members, but do not always have time in the middle of an incident to explain in detail how to perform those tasks.
- Incident response involves an interesting dichotomy of extremely focused analysis work and a need to maintain a constant communication between responders. This is one reason IR teams often use a "war room" with LCD projectors, whiteboards and flip charts. However, when an incident or team spans distant locations, this high-touch level of communication becomes more difficult.
- Due partly to a variety of factors (communication issues, working styles, and lack of shared storage), incident response teams often duplicate work or information. Ensuring team members cross-check each others' work can also be difficult, despite the value, because the teams are often under pressure to produce results as quickly as possible.
- Incident workflow process, even when it already exists, is easy to forget when the team is working furiously to follow up on leads. Afterwards, this can lead to difficulty retracing steps or putting together the details of the response work.
- Communication and collaboration between team members is necessary, but can be a distraction when work has to be interrupted while figuring out how to share information.

These and a number of other observations made during a large scale incident led the author to the creation of Orion.

3. Orion Design Goals

Orion has a few simple design goals. These are to provide:

1. Standard incident response workflow

2. Secure communication, collaboration, and data sharing
3. Consistent data collection
4. Pre-installed tools & scripted analysis
5. Common report formats
6. Incident data and communication archive

Although these design goals were developed as a result of challenges identified during an incident involving the author and his colleagues, they strongly resemble similar findings by the authors of the Palantir system (Khurana, Basney, Bakht, Freemon, Welch, & Butler, 2009).

4. How Is Orion Different?

There are many bootable live security distributions and virtual machine implementations. Some of them provide solutions to a subset of the Orion design goals. Some of the better known distributions (especially ones that provided inspiration) are described here for comparison purposes.

4.1. FIRE

FIRE was created by William Salusky as one of the first bootable CDROM distributions designed specifically for forensics and incident response (Salusky, 2004). FIRE is no longer actively maintained, but many incident responders still carry a copy of the CDROM because it has some tools not easily found elsewhere. FIRE does not meet the collaboration and work flow requirements of Orion.

4.2. HELIX3

Helix3 started as a tool for creating forensic images and was first publically released in 2003 (e-fense, Inc., 2009). The distribution grew to include a large number of open source tools and provide both a bootable Live CD as well as Live Response environment for Windows and Linux. Eventually Rob Lee incorporated into the SANS forensics training track. In 2009, e-fense created Helix3 Pro as a paid, subscription-based product.

After some questions from the community, e-fense also made the last free version (Helix3 2009R1) available, but no further updates will be made.

4.3. SANS Incident Response and Forensic Workstation (SIFT)

For several years the SANS SEC 508 class was taught using the Helix3 CD and virtual machine images. When e-fense went commercial with Helix3, SANS built a customized Fedora-based VMWare image.

> *"Faculty Fellow Rob Lee created the SANS Investigative Forensic Toolkit(SIFT) Workstation featured in the Computer Forensic Investigations and Incident Response course (FOR 508) in order to show that advanced investigations and investigating hackers can be accomplished using freely available open-source tools."* (Lee, 2010)

SIFT not only provides tools, but also additional virtual hard drives that can be used by students to practice acquisition and analysis skills. This is not only an excellent learning tool, but a solid workbench for forensic analysts and incident responders. Although it contains sshfs for encrypted file sharing as well as report writing tools, it doesn't have the focus on team work and collaboration that Orion has been designed with.

4.4. BackTrack

BackTrack has become one of the most sophisticated security distributions ever created – and one of the most actively updated (Offensive Security, 2010). A favorite of security professionals and penetration testers, it contains many offensive tools and some forensic acquisition tools, but does not have an incident response focus. Orion is based on BackTrack because of the modular design and easy customization of the distribution. This was true in version 3, but now that BT4 is based on Ubuntu, adding and removing packages using the Advanced Packaging Tool (apt-get) is extremely easy.

4.5. NST (Network Security Toolkit)

The Network Security Toolkit, NST, is a very comprehensive suite of tools with a sophisticated set of web-based documentation (Network Security Toolkit (NST v2.11.0),

2009). While NST is not focused specifically on incident response, it does include many tools for that purpose and provides some unique capabilities for secure communication – such as creation of ppp-over-ssh tunnels for communication between NST hosts. NST has different installation modes that let the same distribution be deployed as a management server, a sensor, or a number of other configurations.

4.6. Securix-NSM

Securix-NSM is a Live CD based on the principles of Network Security Monitoring (Securix-NSM, 2010). NSM is a methodology that relies on situational awareness, expertise of the analysis team, and gathering as much network traffic data as the organization will allow. Securix-NSM provides network traffic sensors using snort as well as management servers and clients using Sguil, an aggregation tool for the network monitoring data.

4.7. DEFT Linux

DEFT Linux is a Linux 2.6.31 based Computer Forensics Live CD with a variety of file and network forensics tools, including the Xplico graphical network traffic analysis tool (DEFT Linux - Computer Forensics live cd, 2010). DEFT has an attractive user interface based on LXDE (the "Lightweight X11 Desktop Environment"). The DEFT web site also provides the DEFT Extra Computer Forensic GUI. This tool is similar in concept to the HELIX bootable Windows toolkit. It provides imaging and analysis tools that can be directly run from read-only media.

4.8. HeX System 2.0

HeX is rather closely aligned with the design goals of Orion. Its focus, however, is also on Network Security Monitoring – generally pre-incident analysis work – rather than incident response (geek00l, 2010). It does include some collaboration tools, such as the Pidgin instant messaging client and a lightweight IRC client. However, there is no root password and no additional hardening, so HeX is clearly intended to be used in trusted environments.

John Jarocki, john.jarocki@gmail.com

4.9. Palantir

Palantir is a product of Palantir Technologies. The original inspiration for Palantir was *Incident 216* as documented in the original Palantir report (Khurana, Basney, Bakht, Freemon, Welch, & Butler, 2009). Palantir appears to be a very excellent analysis and collaboration platform, but is not based on bootable media and is not an open source framework.

5. What is Orion?

Orion is a customized version of BackTrack 4 that adds tools to support the secure collaboration, incident tracking, and analysis goals of the project. BackTrack was chosen as the base for Orion because it is a respected Linux-based security distribution that comes pre-installed with a number of security tools. Since BackTrack 4 is now based on Ubuntu Linux, it is easy to customize using the aptitude installer and install using tools such as remastersys and ubiquity (Ubuntu Home Page, 2010).

The packages listed in Appendix B were removed because they are intended primarily as attack tools. Other tools have been added to support the analysis and collaboration goals. These full list of tools added to create Orion are listed in Appendix A. Some of the more important ones include:

- Collaboration Tools:
  - Citadel: complete and feature-rich groupware server
  - ssvnc: VNC viewer sessions tunneled over SSH/SSL
  - X11vnc: VNC server for real X displays
- Case Tracking:
  - Incident Response Questionnaire (custom):
    A web-based form to gather details, help triage, build the team, and create a new incident ticket
  - RTIR: Request Tracker for Incident Response
- Network Analysis:
  - argus: IP Network transaction auditing tool

John Jarocki, john.jarocki@gmail.com

  - chaosreader: Fetch application sessions from pcap data
  - dnstop: Display tables of DNS traffic from pcap data
  - sancp: Security Analyst Network Connection Profiler
  - tcpstat: Network interface statistics reporting tool
  - rumint: Network visualization tool for live or recorded pcaps
  - xplico: Network forensic tool
- Situational Awareness:
  - arpalert: monitor ARP changes in ethernet networks
  - tcpick: TCP stream sniffer and connection tracker
  - labrea: a "sticky" honeypot and IDS
  - etherape: graphical network monitor modeled after etherman

Additionally, custom scripts, documentation, and templates have been written and are located in the directory **/orion/** in the Orion file system.

6. Using Orion

Although Orion can be used merely as a collection of tools, the intent is to enforce a consistent workflow. Orion helps the incident responder determine which workflow is appropriate, sets it up, and then encourages its use.

Orion is built around the idea that the team may grow beyond the initial responder. The first responder (or lead responder takes the title “alpha”). Subsequent added team members use the titles bravo, charlie, delta, etc. This is useful for several reasons. First, in a need-to-know environment, the titles can be used to obfuscate the parties involved. It also gives an expectation of consistency – handler alpha is the initial and primary responder.

In the degenerate case, alpha will be the only responder and Orion will not be used for team collaboration. However, secure communication channels can still be created when needed, and the workflow features of Orion are still available. Orion also has a very complete set of analysis and response tools, so even as a standalone system – it holds its own.

John Jarocki, john.jarocki@gmail.com

Most of the "glue" scripts that extend the base BackTrack installation to provide Orion's functionality are stored in **/orion/scripts/.**

6.1. Startup

Orion startup is very similar to BackTrack. At the login prompt, the responder logs in as root, using the initial password. After login, the **startx** command will start a graphical X11 session and the KDE window manager.

```
Orion Alpha orion tty1
based on BackTrack 4 PwnSauce

orion login: root
Password:

Last login: never
Orion Alpha, based on
BackTrack 4 (PwnSauce) Penetration Testing and Auditing Distribution

root@orion:~# startx
```

Figure 1: Initial Orion startup

After the KDE desktop appears, the Firefox panel menu button will open the Orion start page as the default home page.

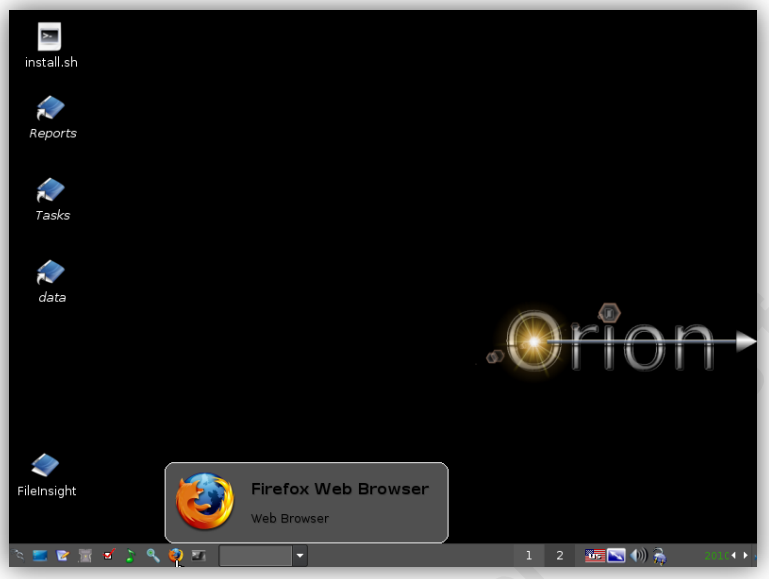

Figure 2: Orion KDE desktop

This site serves as a memory aid to guide the user through the incident response process. The responder would typically move from left to right through the web site tabs as the incident progresses.

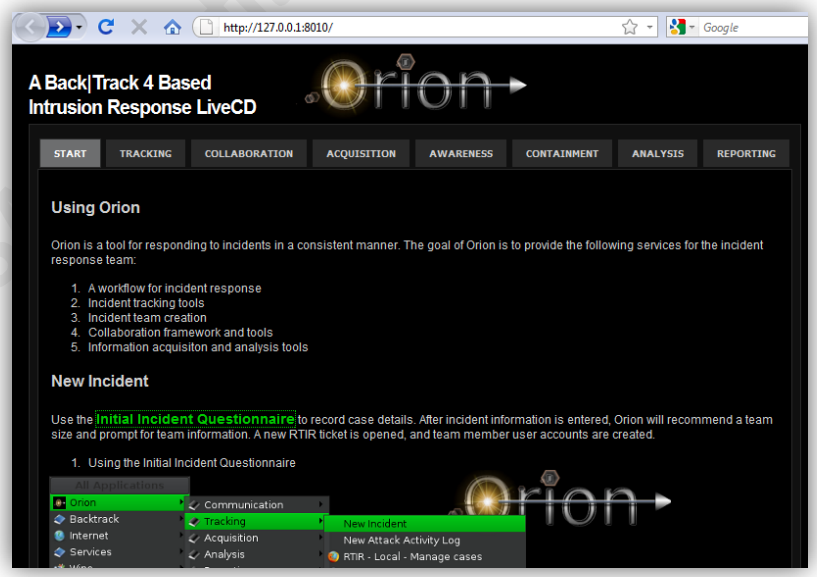

Figure 3: Orion start page

John Jarocki, john.jarocki@gmail.com

Alternatively, the KDE "K-Menu" and panel buttons can be used to navigate the tools included with Orion.

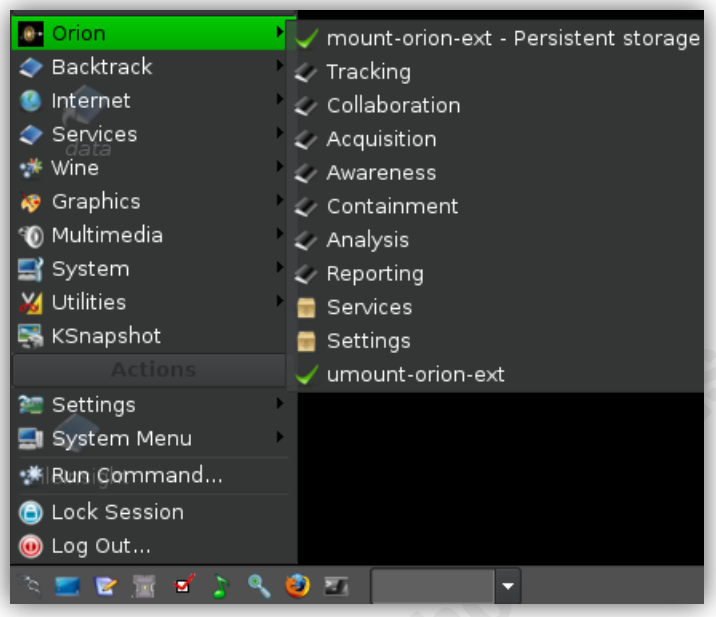

Figure 4: Orion K-Menu

The menu items are arranged in the same order as the Orion web site, and the panel buttons are arranged so that the first tasks are to the left. The new incident questionnaire button is the second one on the panel. Of course, the user can re-arrange these to suite his or her liking.

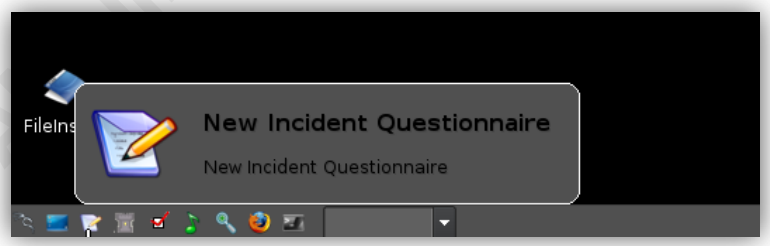

Figure 5: New Incident Questionnaire panel button

John Jarocki, john.jarocki@gmail.com

6.2. Initial Incident Entry

After the startup, the handler enters the details of the new incident. This information can either be collected using the Initial Incident Questionnaire (recommended), or entered directly into ITIR – the Incident Tracker for Incident Response tool.

After incident information is entered, Orion will recommend a team size and prompt for team member contact information. The questionnaire captures this information and saves it for use in various other tools.

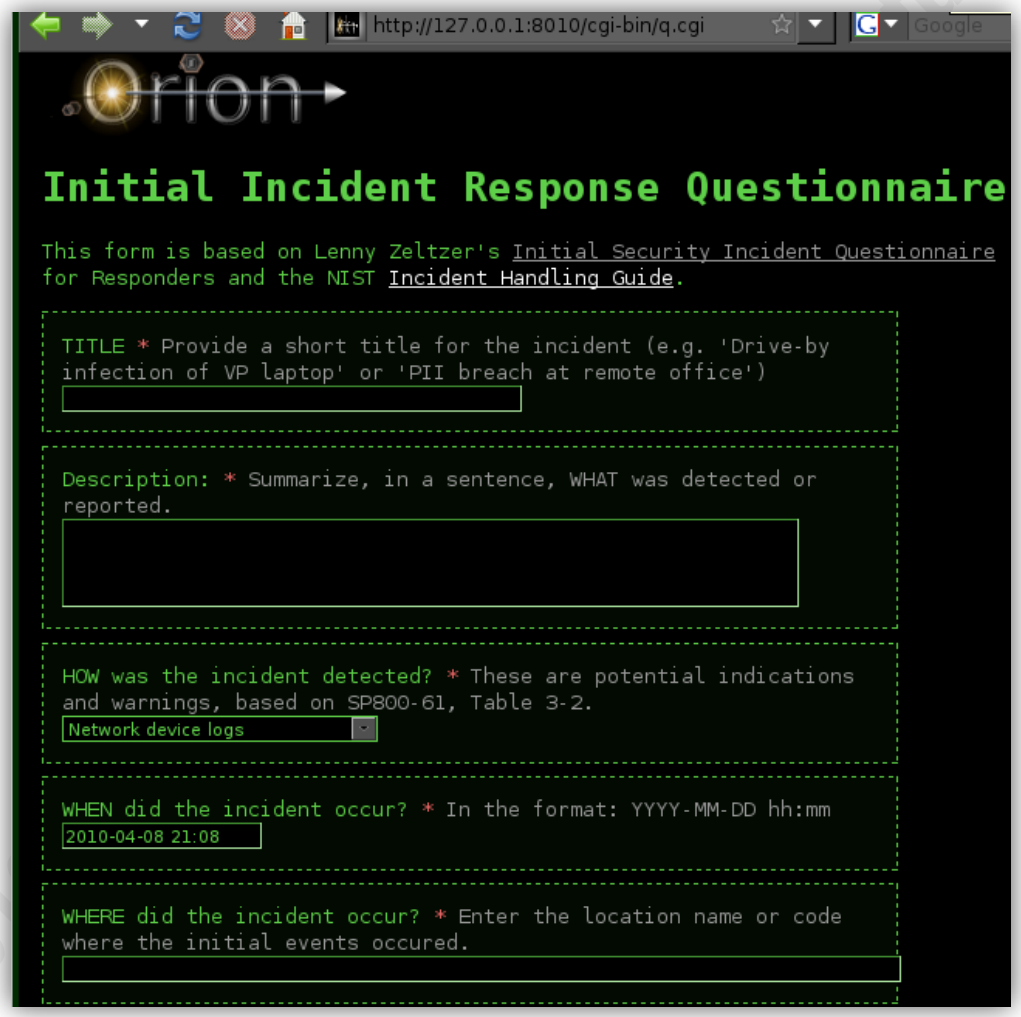

Figure 6: Incident Questionnaire

John Jarocki, john.jarocki@gmail.com

6.3. Incident Tracking

For entry and tracking of incident data, Orion utilizes the Request Tracker for Incident Response (RTIR) tool from Best Practical. When the incident responder enters initial case information via the Incident Response Questionnaire, the final page of that form-based system creates the following:

1. Citadel user accounts for the Incident Response Team members,
2. RTIR accounts for the team members, and
3. A new RTIR **Incident Report** ticket.

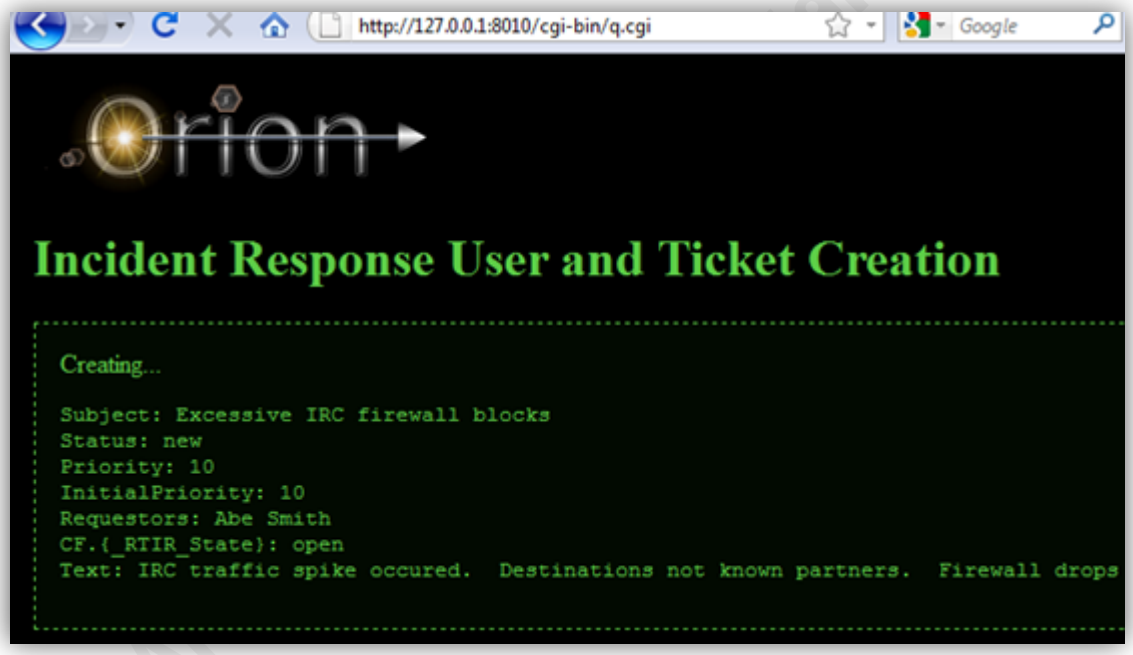

Figure 7: New RTIR Incident Report ticket

RTIR (Request Tracker for Incident Response) is a full-featured open source case tracking tool that was designed specifically for computer security incidents. According to the Best Practical web site, the developers, "worked with over a dozen CERT and CSIRT teams to build a world-class incident handling system." (Best Practical, 2010)

RTIR is a modified version of the original RT that was customized to create an incident response workflow. For responders who are not familiar with the RTIR workflow, it can

take some time to get used to it. As shown in the diagram, new information about a potential incident is entered first as an **Incident Report**. Once the report has been verified, an **Incident** is created. This incident can then lead to one or more **Investigations** as well as **Blocks**. A Block is an activity that attempts to stop or retard the attacker activity. Blocks are essentially containment steps. The investigations could be analysis work, interviews, forensic acquisitions, or any other activity that leads to the understanding and resolution of the incident. A visual depiction of the relationship between these entities is seen in Figure 8: RTIR Workflow.

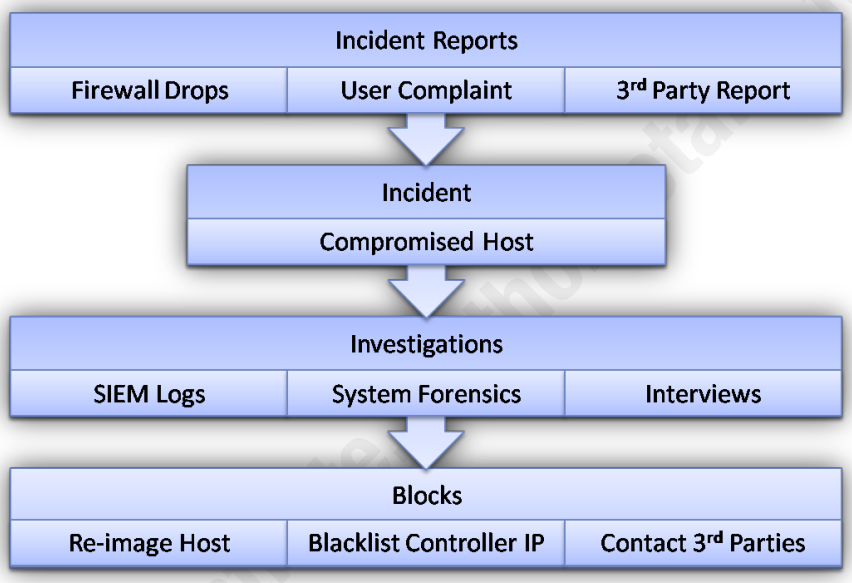

Figure 8: RTIR Workflow

6.3.1. Using RTIR

The incident responder can use the RTIR web based interface (http://127.0.0.1/) to login and perform a variety of actions, such as accepting, annotating, rejecting, and closing incidents.

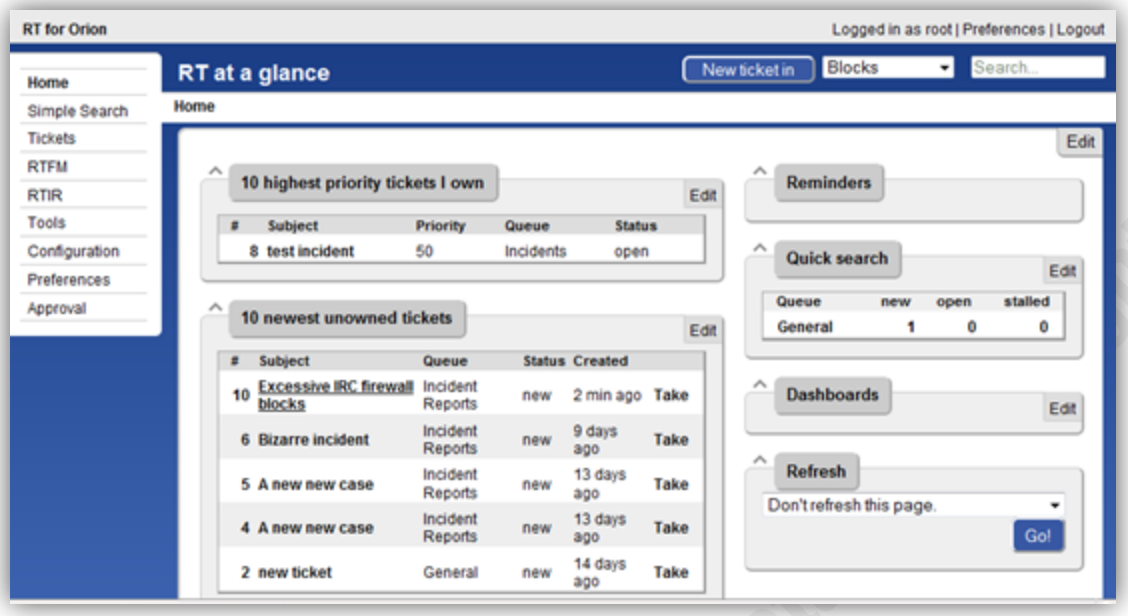

Figure 9: Initial RTIR site

In this example, the information entered in via the Initial Questionnaire was created as an Incident Report automatically.

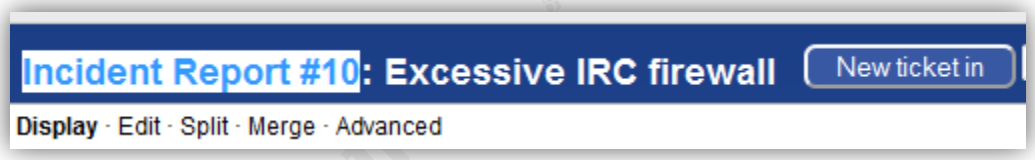

Figure 10: RTIR Incident Report

The next screenshot shows the creation of a new Incident based on the incident report. After the incident has been created, subsequent related incident reports can be linked to it.

John Jarocki, john.jarocki@gmail.com

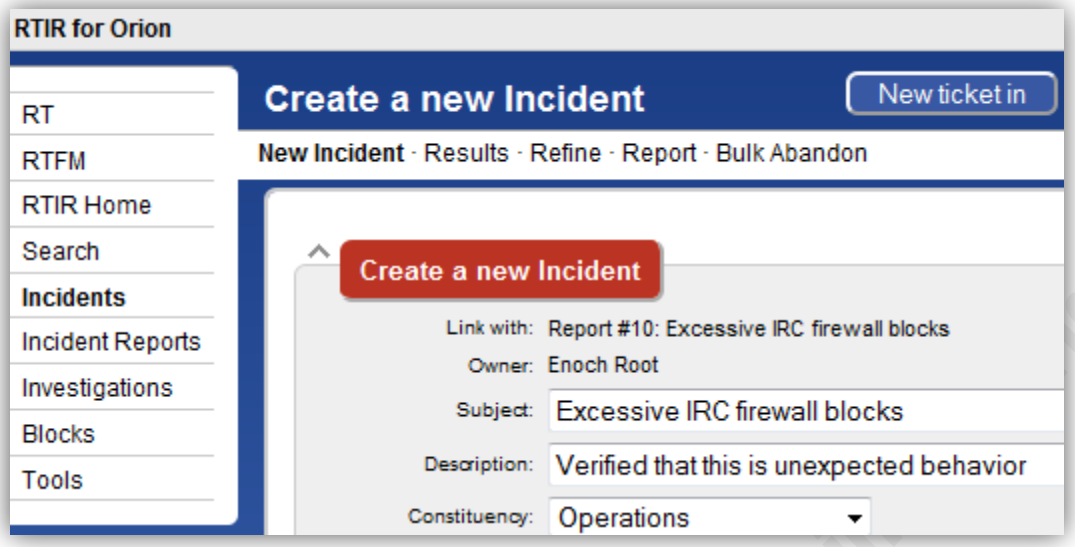

Figure 11: RTIR Incident

The installation of RTIR in Orion stores incident information in a mysql database. This database can be accessed and manipulated using any tools that support mysql. For example, the **mysqlhotcopy** command can be used to make a backup of the RTIR database:

```
mysqlhotcopy --user=<userid> --password=<password> rt3 /backup
```

Future versions of Orion will scripts to automate the backup, validation, and restoration of the RTIR database. This will allow case tracking data to be archived for a specific incident. The released version of Orion will include a script to archive the RTIR database and any other data related to the incident.

6.4. Secure Communication

Communication during incident response must be trustworthy, but it also needs to be full-featured. Attackers can hamper the ability of the incident handling team to provide a timely response simply by creating doubt in the trustworthiness of normal communication channels. If the email system has been compromised, how does the team keep abreast of the situation? If normally used instant messaging systems are

John Jarocki, john.jarocki@gmail.com

unencrypted in the presence of an active attack, a packet capture can undermine all defensive efforts.

Orion provides a trusted communication framework that is configured in a secure fashion, and uses encrypted protocols to tunnel communications between team members. At the same time, the team can share desktop sessions, voice, and chat sessions. Many tools and protocols have built in encryption for communication. However, Orion tries to keep this simple by using the secure shell (SSH) protocol to tunnel all traffic. In this way, the Orion users can know that normal traffic will always be SSH between the known responder systems and other traffic is automatically abnormal and suspicious. As long the protocol can be tunneled over SSH, it can be used in Orion.

Orion includes a number of tools (and leverages ones already found in BackTrack) to assist the responders in the setup of secure communications.

- **new-ssh-key**: creates a new RSA2 key pair. Puts the private key in ~/.ssh/orion and the public key in ~/.ssh/orion.pub. Use a strong passphrase. The passphrase will be used to encrypt the private key with 3DES.
- **copy-ssh-key**: Copies the orion pub key to a remote host and places it in ~/.ssh/authorized keys. Uses ssh-copy-id from BackTrack.
- **proxy-start**: Create a proxy tunnel using ssh -D localhost:9050 <user>@<host> The tunnel will listen on localhost:9050, making it compatible with TOR and proxychains.
- **proxychains**: is already installed in BackTrack, and therefore Orion. Once proxy-start has been executed, a command like "proxychains ssh <otherhost>" will proxy an SSH login to <otherhost> via <host>.
- **orion-tunnel**: uses SSH to allow the responder to log into the primary handler's machine and tunnel the list of services typically supported by Orion.

John Jarocki, john.jarocki@gmail.com

6.5. Collaboration

The **orion-tunnel** command uses SSH port forwarding to forward a short list of services to the primary responder's system. For example, if the secondary responder runs orion-tunnel and then opens the URL http://localhost/ in a web browser, it will display (via encrypted SSH tunnel) the RTIR server web interface from the primary system.

Port forwarding works by initiating an SSH session with one or more -L command line arguments, such that **-L lport:rhost:rport** forwards connections from port **lport** on localhost to port **rport** on the host **rport**. One could also use dynamic port forwarding (the -D option) to forward all connections using the SOCKS protocol, but the connections between handler systems are kept to a minimum, by design, to avoid any unwelcome surprises and also to make connections to 127.0.0.1:<port> do "the right thing" automatically whether the primary handler is local or remote.

Server	Port (protocol)
Orion web site	8010 (http)
Citadel	25/465 (smtp/s), 8504 (http), 993 (imaps), 5222 (xmpp)
RTIR web interface	80 (http)
Xplico	9876 (http)

6.5.1. Shared Desktop

Desktop sharing is also an extremely valuable capability during incident response. In Orion, desktops can be shared using VNC (Virtual Network Computing) tunneled over SSH. The x11vnc program allows the X11 system to share display zero (:0). This means a remote VNC session will connect to the console being used by the local user. The remote user then uses the Enhanced TightVNC tool, SSVNC (http://www.karlrunge.com/x11vnc/ssvnc.html), to tunnel a connection to the VNC session over SSH.

John Jarocki, john.jarocki@gmail.com

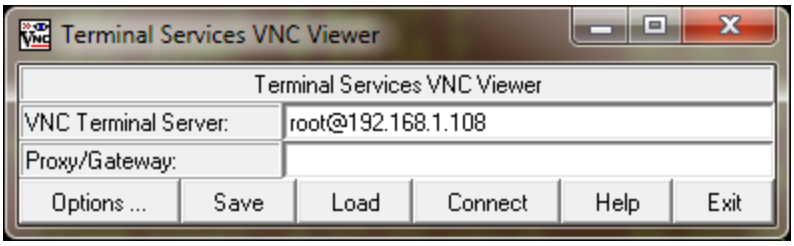

Figure 12: tsvnc

The ssvnc tool runs on many platforms. It is included in Orion so responders can share other Orion desktops. It can also be installed on other platforms, such as Windows to allow collaboration with people who are not using currently using an Orion system.

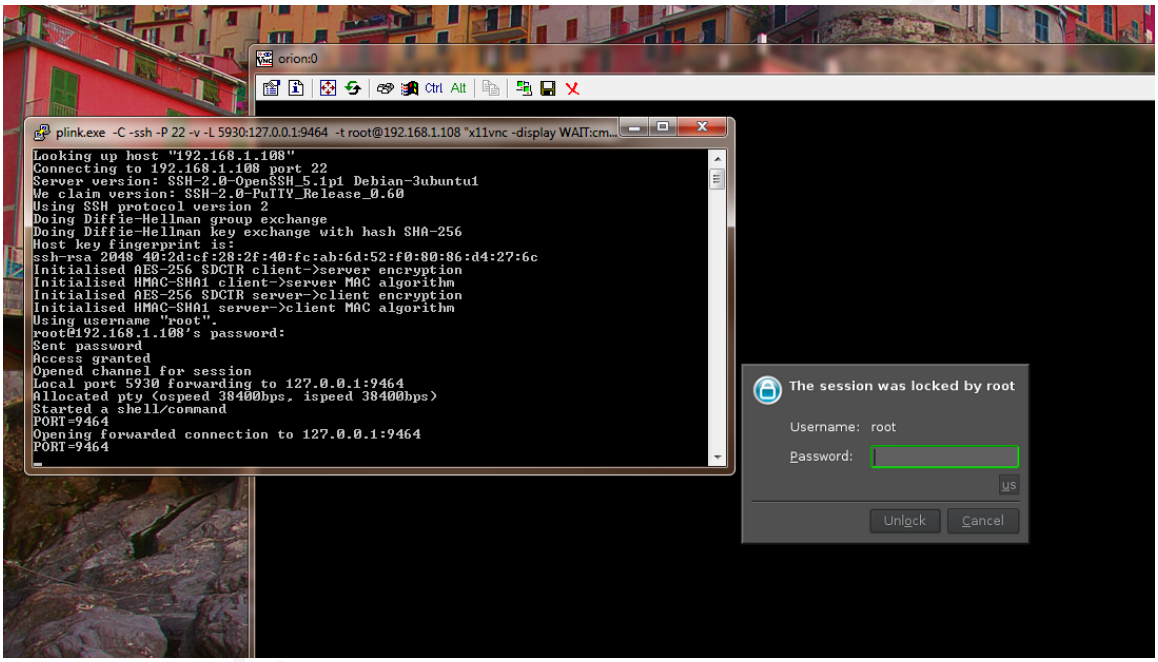

Figure 13: Shared VNC session over SSH

6.5.2. Shared File Systems

Orion uses sshfs (FUSE file system support for SSH) to provide an encrypted tunnel for shared file systems between handlers. The home directory of every handler has a mount point for every other handler. In this way, analysis data can be shared, securely, in a real-time fashion between team members. One side note is that sshfs does not accept the path to the private key as a command line argument, so the following line must be added to the /etc/ssh/ssh_config file:

```
IdentityFile ~/.ssh/orion
```

```
/orion/scripts/mount-ssh:
 mount remote directory as a filesystem via ssh
================================================
Remote host or IP: remhost
Remote username: john
Remote path [~]: kindle
Path of local mount point [e.g. /mnt/alpha]: /mnt/kindle
Mounting john@remhost:kindle at /mnt/kindle ...
Enter passphrase for key '/root/.ssh/orion':
Filesystem                    1K-blocks      Used Available Use%
Mounted on
john@remhost:kindle                   0        -0         0 103%
/mnt/kindle
Hit any key to continue ->

root@orion:~# ls /mnt/kindle
download_pdfs            kindle_update_tool.py   extract_bin.txt
kindle_update_tool.zip   mobilereader.com.txt    restart_log_reset.txt

root@orion:~# mount | grep kindle
john@otherhost:remhost on /tmp/kindle type fuse.sshfs (rw,nosuid,nodev)
```

Figure 14: sshfs in action

6.5.3. Citadel: Groupware for Incident Communication

For groupware collaboration (i.e., email, calendar, chat) Orion includes an installation of Citadel (http://www.citadel.org/). Citadel is a feature-rich groupware server that started as an attempt to replicate a BBS system back in the days before the World Wide Web. In the intervening years, it has grown mature and sophisticated, and the web based interface to Citadel (webcit) is a slick AJAX-based environment. It provides email, calendar, chat and other services via text-based, web-based, and standard protocols such as SMTP, POP, IMAP, and XMPP. It is intended to be run on the Orion system of the primary handler, but an instance could be run on every Orion system, if so desired.

John Jarocki, john.jarocki@gmail.com

Figure 15: Citadel login screen

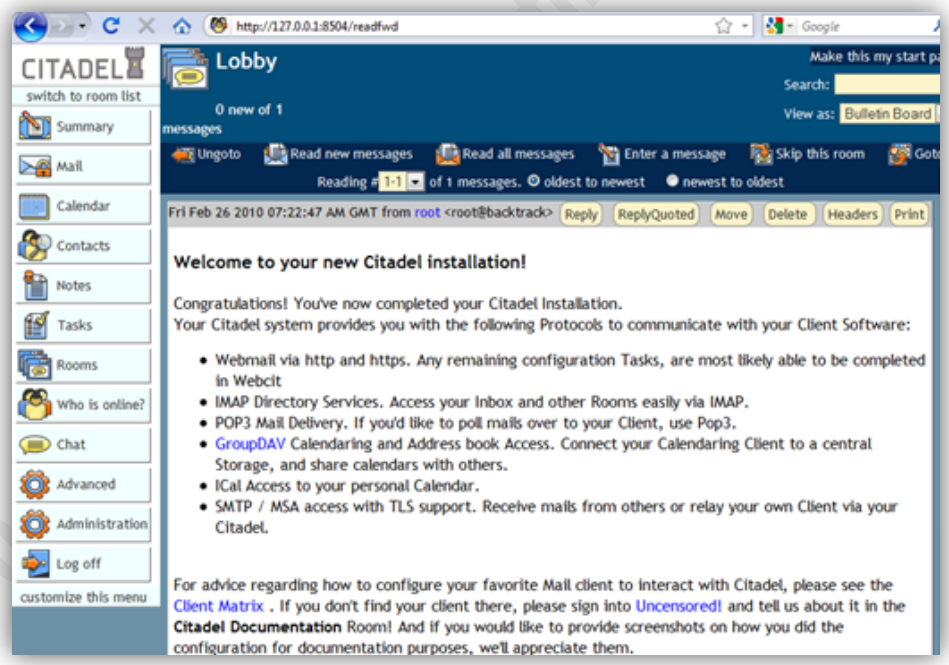

Figure 16: Citadel welcome screen

Citadel gives the team a complete collaboration environment, with an email server, calendar, contact database, task list, and chat rooms.

John Jarocki, john.jarocki@gmail.com

True to its roots, the Citadel server still has a completely functional command line interface reminiscent of the bulletin board systems (BBS). The **citadel** command initiates this text-based interface.

```
alpha@orion:~/$ citadel
Citadel 7.37
Orion

This system is soley for use by authorized users for official purposes. Users
have no expectation of privacy.  Use of this system constitutes consent to
monitoring, retrieval, and disclosure of any information stored within for any
purpose including criminal prosecution.

Enter your name: alpha
Please enter your password:

Lobby> Who is online
   User Name                Room              Idle    From host
 ------------------------ ------------------ ---- --------------------
   alpha                    Lobby                    orion
 ------------------------------------------------------------------------

Lobby> Chat
Entering chat mode (type /quit to exit, /help for other cmds)
```

Figure 17: Citadel's old school interface

Finally, Citadel can speak standard Internet protocols such as Simple Mail Transfer Protocol (SMTP), Internet Message Access Protocol (IMAP), and Extensible Messaging Presence Protocol (XMPP) – sometimes referred to as the "Jabber protocol." With this capability, team members can use email and chat clients of their choice.

The ability to access the Citadel server information from three different interfaces (web-based, text-based, and native protocol) allows the team members to communicate in whatever form is available, convenient, and desired for them at the time. Orion secures these communication channels via SSH encrypted tunnels between responder systems. The Orion default firewall configuration only allows SSH traffic, so connections to the Citadel services must be authenticated and encrypted from remote systems. In practice, the SSH tunnels to these services can also be created from other systems – such as the Windows 7 client used to take the screen shots in this paper.

John Jarocki, john.jarocki@gmail.com

6.5.4. Citadel Data Archive

The recommended configuration is a Citadel server on the primary handler's system with periodic backups of the database. Orion includes the script **backup-citadel** to copy the Citadel data and configuration to a backup location using rsync over SSH. The Citadel database is contained within the directory /var/lib/citadel, and the remaining configuration can be found in /etc/citadel. Citadel can also maintain complete transaction logs in /var/lib/citadel/data/. With this feature enabled, an off-system archive of this data provides a very handy forensic history of all communication and collaboration associated with the incident.

```
root@orion:/orion/scripts# ./backup-citadel
Usage: ./backup-citadel <path>
Rsync Citadel data and config to <path>
=======================================
Path (user@host:/path if remote): john@192.168.1.101:ORION
Enter passphrase for key '/root/.ssh/orion':
building file list ... done
created directory ORION
citadel/
citadel/bio/
citadel/bitbucket/
citadel/data/
citadel/data/cdb.00
citadel/data/cdb.01
citadel/data/log.0000000001
citadel/files/
citadel/images/

sent 10714110 bytes  received 426 bytes  1020432.00 bytes/sec
total size is 10711559  speedup is 1.00

Hit any key to close this window ->
```

Figure 18: backup-citadel command

John Jarocki, john.jarocki@gmail.com

6.6. Data Acquisition

Acquiring data related to an incident is a task that needs to be done quickly, but also carefully. Digital forensics experts recommend retrieving data in the order of volatility (Henry, 2009). There are many forensic imaging software packages. Some of these are included in Orion, and others can be installed. Orion also includes a simple script, described below, for acquisition of volatile Windows system information.

6.6.1. Remote Files and Logs

In order to analyze logs and malware samples, Orion provides several scripts for acquiring the data from remote hosts. The remote hosts might be other analysts' systems, or they might be compromised systems. In the latter case, caution needs to be exercised when passing credentials. The current version of Orion attempts to use encrypted protocols and encourages the use of SSH key pairs, but it otherwise provides no mitigation against remote services replaced with trojan software. This is a topic for future research.

The **acquire-path** script can be invoked from the Orion menu or command line to retrieve files from a remote host. The script uses scp if the path specifies a remote host name or cp if not. The latter case is useful if the remote file system has already been mounted locally with sshfs, SMB, or other protocols.

John Jarocki, john.jarocki@gmail.com

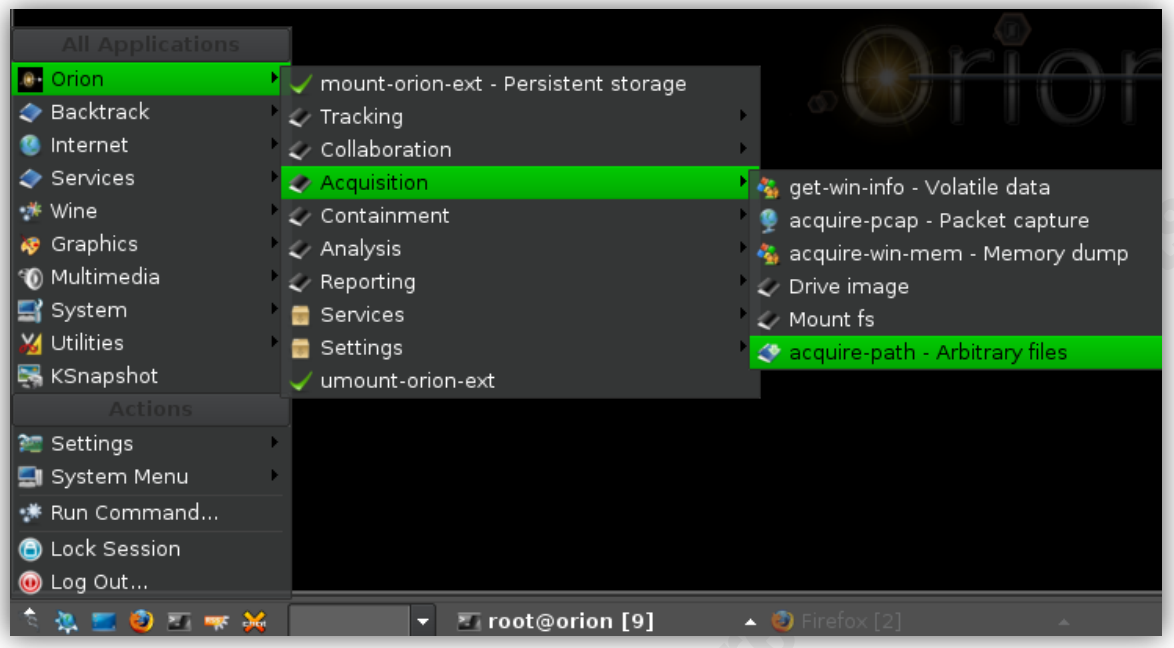

Figure 19: Orion acquire-path menu item

The **acquire-path** tool is simple, but it enforces a standard methodology for transferring data to the responder's system in a repeatable way. It uses, encrypted file transfers and an SSH key pair for authentication when the public key has been distributed to the remote host. The example below also shows the MD5, SHA1, and SHA256 hashes that are stored in hashes.md5.txt, hashes.sha1.txt, and hashes.sha256.txt in the data store. The timestamp stored when the hashes are written allows multiple copies of the same data to be retrieved at different times. The hashes can be compared to detect changes.

```
root@orion:~# /orion/scripts/acquire-path john@host:/tmp/a.pcap
/usr/bin/scp -i /root/.ssh/orion -r john@host:/tmp/a.pcap
/root/data/tmp
Enter passphrase for key '/root/.ssh/orion':
a.pcap                                100%    0     0.0KB/s   00:00
# Sun Apr 11 01:19:40 BST 2010
d41d8cd98f00b204e9800998ecf8427e  /root/data/tmp/a.pcap

Hit any key to close this window ->
```

John Jarocki, john.jarocki@gmail.com

Notice that the file a.pcap was zero bytes in length in the first transfer and has grown in this second example to 28 bytes and now has different hashes.

```
root@orion:~# /orion/scripts/acquire-path john@host:/tmp/a.pcap
/usr/bin/scp -i /root/.ssh/orion -r john@host:/tmp/a.pcap
/root/data//tmp
Enter passphrase for key '/root/.ssh/orion':
a.pcap                              100%   28     0.0KB/s   00:00
# Sun Apr 11 01:20:29 BST 2010
663863b054d8792e6f3ab1249d5a4f7f  /root/data/tmp/a.pcap

Hit any key to close this window ->
```

This can be seen by viewing the hash files themselves. If a file is being collected repeatedly via a cron job, for example, the change to the hashes can be used to alert the responder of a new condition. In this way, Orion can be used as a detective tool.

```
root@orion:~# tail data/md5.hashes
# Sun Apr 11 01:19:40 BST 2010
d41d8cd98f00b204e9800998ecf8427e  /root/data/tmp/a.pcap
# Sun Apr 11 01:20:29 BST 2010
663863b054d8792e6f3ab1249d5a4f7f  /root/data/tmp/a.pcap
```

In some cases, the data to be acquired will be mounted already on the responder's workstation. The file system might be mounted using sshfs or SMB/CIFS from a remote system. The other possibility is that a read-only copy of an acquired drive might be mounted using loopback (**`mount -o ro,loop image.dd /mnt/dd`**) or the vmware-mount.pl utility (http://www.vmware.com/support/reference/linux/loopback_linux.html) in the case of VMware disk images (vmdk).

When the data is mounted, the responder can use **acquire-path** to retrieve a subset of the data, place it in the data area, and perform the same hashing that is performed when acquiring data from a remote system. Since **acquire-path** uses the bash readline function

to read the path information from the user, tab auto-completion and command line editing are supported. In the example below, the tab key was pressed twice to list the directories and folders under the path /mnt/smb/.

```
root@orion:~# /orion/scripts/acquire-path
/orion/scripts/acquire-path:
 Recursively copy files in the specified path
 using either scp (remote) or cp (mounted).
 Files and checksums are put in /root/data/.
================================================
Path (user@host:/path if remote): /mnt/smb/
AUTOEXEC.BAT              ntldr
boot.ini                  pagefile.sys
CONFIG.SYS                Program Files/
dbc96c92/                 sans/
Documents and Settings/   System Volume Information/
img.dd                    temp/
IO.SYS                    tools/
MSDOS.SYS                 WINDOWS/
NTDETECT.COM

Path (user@host:/path if remote): /mnt/smb/dbc96c92

/bin/cp -r /mnt/smb/dbc96c92 /root/data//mnt/smb

ed97f3276d9fabcf0068de2172df8da5  /root/data/mnt/smb/dbc96c92/hh.exe
e20fa6287839fb0086e859f265dc74cd  /root/data/mnt/smb/dbc96c92/iti.dll
[ ... ]
```

Copying files from a mounted SMB share

6.6.2. Windows System Information

The script **/orion/scripts/get-win-info** utilizes Andrzej Hajda's winexe program to execute commands to gather volatile information from a remote Windows host.

John Jarocki, john.jarocki@gmail.com

The **run-win-cmd** script can be used to remotely run arbitrary Windows commands. For example, ManTech's Memory Dump Utility mdd.exe can be copied to the Windows target and executed. The output memory dump file can then be transferred to the Orion workstation using SMB or other methods.

```
/orion/scripts/run-win-cmd:
Use winexe to execute command on remote windows host
=====================================================
Remote host or IP: 192.168.1.109
Command: mdd -o img.dd
Remote username [administrator]:
Password for [WORKGROUP\administrator]:
mdd -o img.dd
 -> mdd
 -> ManTech Physical Memory Dump Utility
    Copyright (C) 2008 ManTech Security & Mission Assurance
 -> This program comes with ABSOLUTELY NO WARRANTY; for details use
option `-w'
    This is free software, and you are welcome to redistribute it
    under certain conditions; use option `-c' for details.
 -> Dumping 511.48 MB of physical memory to file 'img.dd'.
 130940 map operations succeeded (1.00)
 0 map operations failed

 took 112 seconds to write
 MD5 is: 17e52ec3c80400ecaf722551fb6ec5a9

Hit any key to close this window ->
```

6.6.3. Drive Imaging

Orion includes several drive imaging tools that come with BackTrack, such as dcfldd and AIR Imager. Windows-based imaging tools, such as the popular FTK Imager can also be used within the WINE Windows emulation environment (http://www.winehq.org/), especially when the raw image has already been acquired and the incident response team is ready to look at the contents.

John Jarocki, john.jarocki@gmail.com

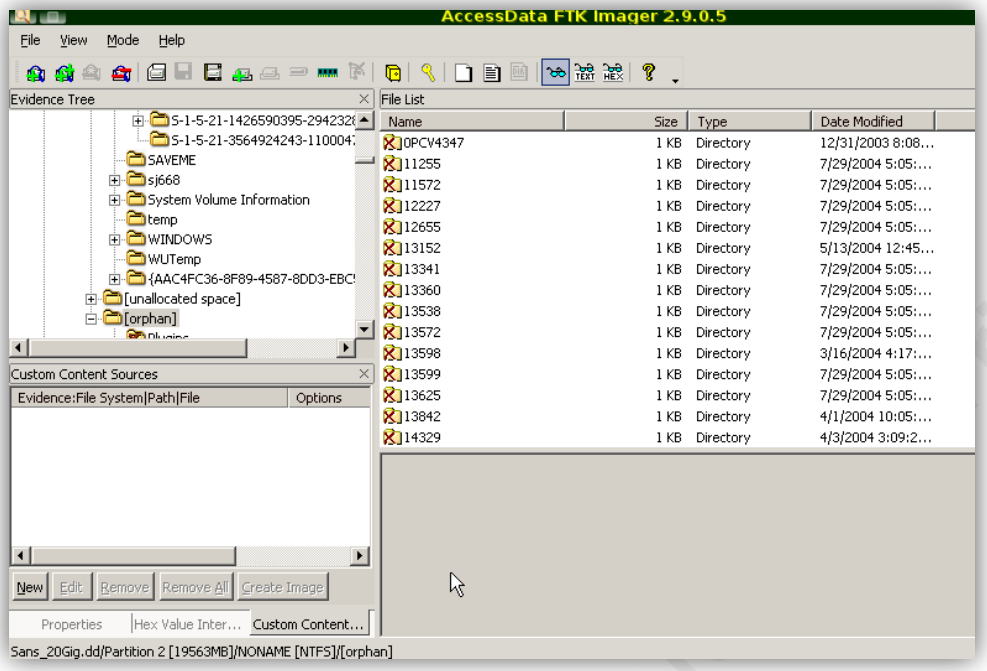

Figure 20: Using FTK Imager in Orion

7. Awareness

Since Orion is intended to be used in the line of fire during an incident, it includes a number of tools that aid in situational awareness for the responder. One such tool, etherape (http://etherape.sourceforge.net/), is a graphical monitor for network activity. Etherape is a quick way to assess the type and direction of network traffic flow on the LAN. Another included tool is arpalert (http://www.arpalert.org/), which watches ARP activity on the local area network and detects any non-whitelisted MAC addresses. It can also be configured to run arbitrary commands to alert the responder or take other actions. Since SSH is the only authorized protocol, Orion also uses the denyhosts (http://denyhosts.sourceforge.net/) tool to blacklist IP addresses that are the source of too many failed SSH login attempts.

The Orion script **tcpick-int** uses the tcpick (http://tcpick.sourceforge.net/) utility for highlighting incoming traffic with color and extracting TCP options and data. This can be run in a small window to watch for unexpected traffic. Here is an example of an unexpected telnet session attempt:

John Jarocki, john.jarocki@gmail.com

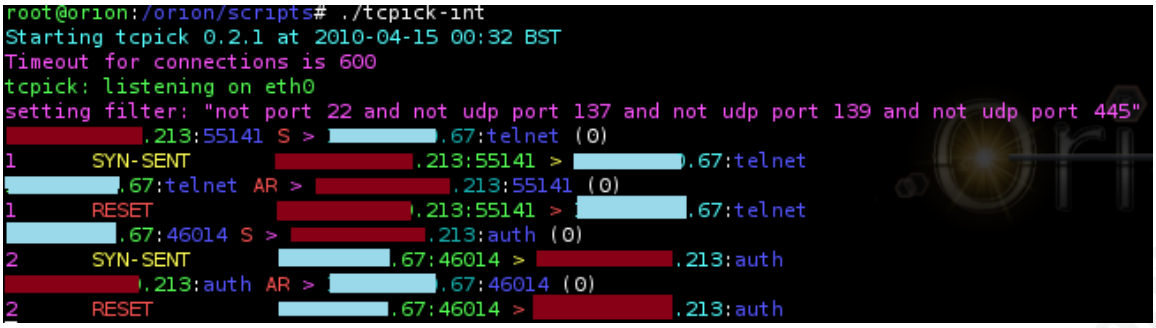

Figure 21: tcpick detects a telnet attempt

Orion also provides a script called **rain** (realtime audible indicator notification) that looks for interesting activity in logs and uses the espeak (eSpeak: Speech Synthesizer, 2010) voice synthesizer to alert the responder. This allows the responder to focus on tasks in progress without having to switch windows to visually keep track of logs. The **rain** tool is currently very crude, but it can be effective to get the responder's attention and can be extended as needed.

8. Containment

Currently Orion's containment capabilities are very rudimentary. Dug Song's tools tcpnice and tcpkill, which slow down TCP sessions (using window size adjustments, et al.) and terminate them (using reset packets), respectively, are included. The labrea tarpit tool, by Tom Liston, is also installed in Orion.

9. Analysis

Orion comes pre-installed with malware analysis tools that run natively in Linux, as well as Windows tools that have been installed in the WINE framework. Because of the inherent danger of analyzing live malware, the recommended use of Orion in this role is to create a separate installation of Orion – preferably virtualized with the ability to create and restore snapshots. The analyst can then use his or her primary Orion system to remotely access the malware analysis system via the desktop sharing, data acquisition, and other tools provided in both Orion installations.

John Jarocki, john.jarocki@gmail.com

10. Malware Analysis

The skill level of malware analysts varies widely. Orion tries to accommodate highly skilled analysts while creating some quick and dirty tools and methods for performing triage. If the analyst is junior, we recommend the analyze-win-malware script mentioned below be used for a quick initial analysis followed by submission to one of the automated online analysis tools, such as the stellar CW Sandbox. Experienced team members will then want to use one or more of the installed debugging and unpacking tools discussed below to perform manual analysis.

10.1. Static Analysis

Orion includes a number of static malware analysis tools. Some of these tools run within WINE to allow the work in an environment that is mostly native to the Orion installation. This figure shows McAfee FileInsight running under WINE to analyze a malicious binary. FileInsight is a great platform for analysis because it has a python-based plugin framework. For example, it includes a plugin for submission of samples to VirusTotal for analysis. Didier Stevens has also written some FileInsight plugins (included in Orion). Future versions of Orion will also have some additional custom plugins that are currently in development.

John Jarocki, john.jarocki@gmail.com

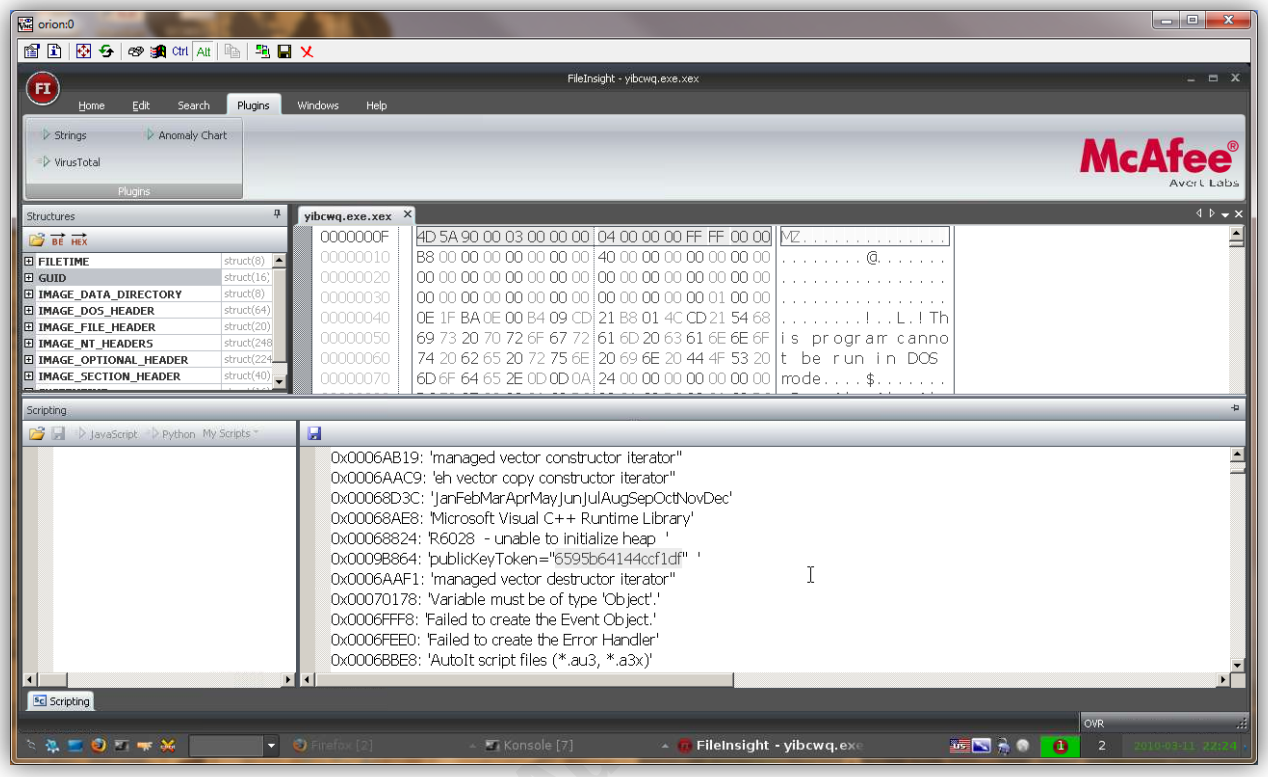

Figure 22: McAfee FileInsight running in Orion under WINE

For a quick review of a suspected bit of malicious data, Orion provides a script called **analyze-win-malware** that currently uses the following tools to perform some quick checks:

- **feagent.exe**: Nick Harbour's tool for detecting packed and armored executables.
- **XORsearch.exe**: Didier Stevens' tool for searching for a string that has been XOR, ROL, or ROT encoded. Orion searches for "http" and "exe" in files.
- **extractscripts.py**: A python script, also written by Stevens, to extract scripts from html files.
- **exiftool**: A perl script written by Phil Harvey that extracts meta data information from a variety of file types.
- **xxd**: Creates a hex dump of the file.

John Jarocki, john.jarocki@gmail.com

Orion additionally has menu items for Didier's FindEvil tool, PDF Parser, and several debuggers such as OllyDBG, Evans Debugger, and IDA Pro Free -- some of which are included because they come with BackTrack.

10.2. Behavioral Analysis

Researchers and responders often use virtualized copies of operating systems running under virtual machine software such as VMWare to perform live analysis of malware.

Orion includes the kernel patches and some scripts to manage VM images, but this cannot be distributed with the Live CD. As a result, the user will have to install VMware after the fact. Also, virtual machine images are pretty large, so the author keeps them located on separate media, and uses symbolic links to inform VMWare where to find them. Some of the recommended virtual machines for this analysis work are:

- Windows XP, with minimal patches
- Windows XP, fully patched
- The Federal Desktop Core Configuration (Windows XP again) available from http://fdcc.nist.gov/
- Fedora and/or Ubuntu Live CD .iso files that can be booted within a VM

This is only a small sample of useful virtual machines for behavioral or dynamic analysis. A complete discussion of this topic is beyond the scope of this paper, but each of these has been tested running as a client OS under Orion.

10.3. Network Analysis

Orion includes several of the author's favorite tools for analyzing network traffic captures. Some of these, such as Xplico (Gianluca & De Franceschi, 2010) are self-contained systems for complete forensic analysis of packet captures (.pcap files). Other tools, such as analog, search log files. In the internal version of Orion in use at the author's organization, the Splunk log reporting and analysis tool is installed. Splunk is a terrific tool for diving into log files – even ones that are completely new to the responder – because Splunk allows for on-the-fly creation of new schemas and field parsers.

John Jarocki, john.jarocki@gmail.com

Unfortunately, the Splunk license does not allow re-distribution, so it will not be available in the public release of Orion.

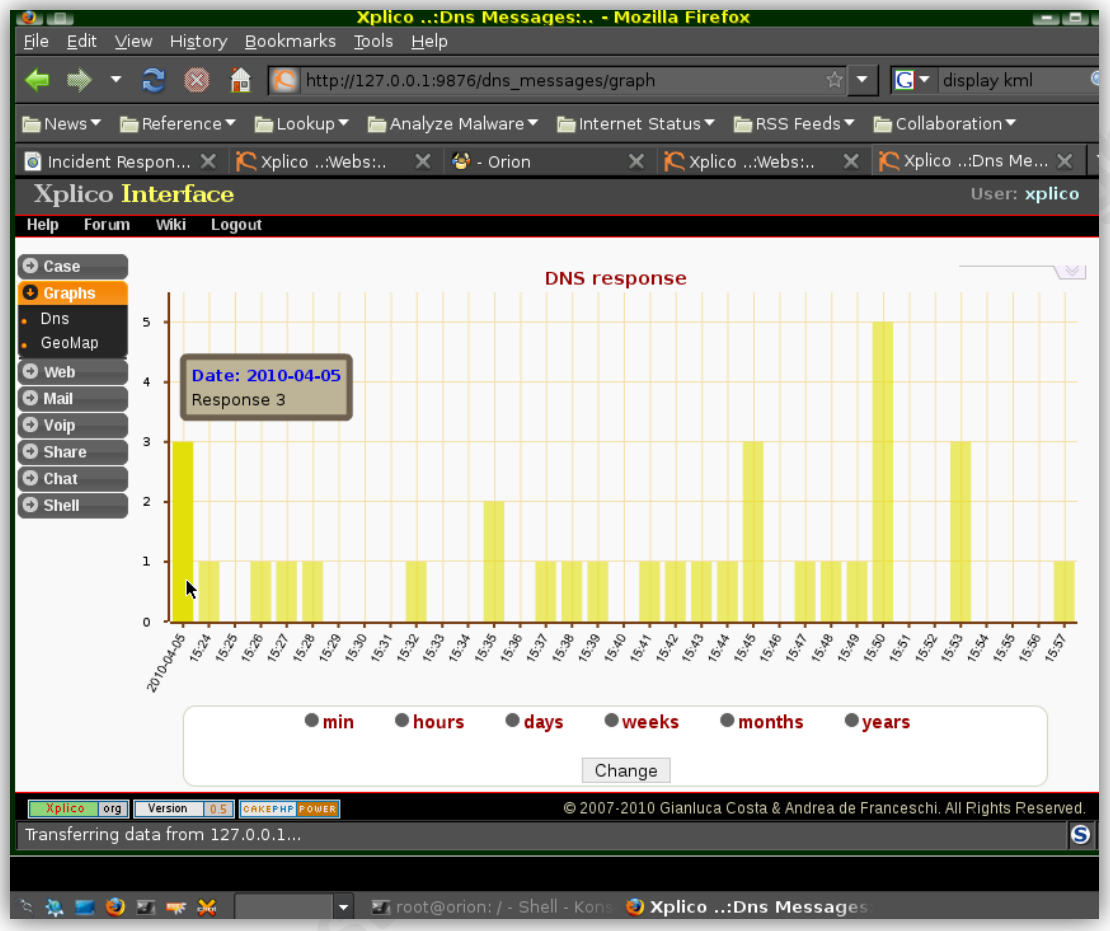

Figure 23: Xplico viewing DNS traffic from an incident pcap

John Jarocki, john.jarocki@gmail.com

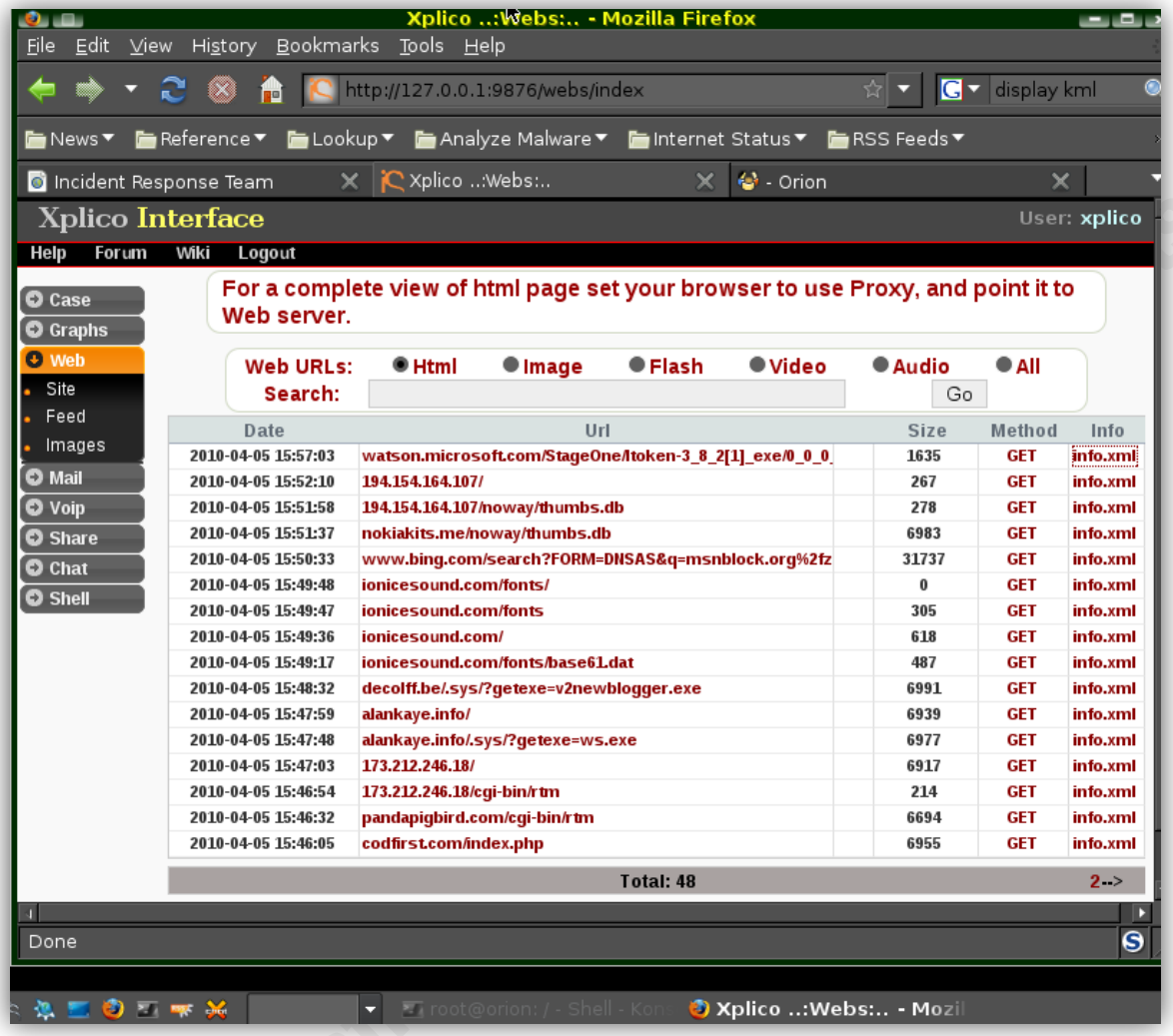

Figure 24: Xplico timeline of HTTP requests from malware

Another excellent tool for ad-hoc analysis of log data is the visualization tool ggobi. The ggobi software takes comma separated value (.csv) or XML data as input and creates multi-dimensional views.

Rumint, the "network VCR player" is another included visualization tool. This tool uses multiple different visualization techniques, that can be combined together to highlight outlying data or interesting patterns.

There are also many, many tools that extract different views of the data contained in .pcap files. Orion includes a script (**/orion/scripts/analyze-pcap**) created by one of our team members that takes a packet capture as input, and produces output from the

following and more: argus, chaosreader, dnstop, dsniff, ettercap, snort, tcpflow, tcpick, tshark. These tools were chosen from a combination of personal analyst experience and tips from other researchers gleaned from the web ("Tshark examples", 2010). The output of those tools quickly provides the analyst with TCP session info, DNS lookup statistics, HTTP headers, extracted files, and other valuable information. Since this is done by a script, the same commands are run each time, and junior members can do the work without a lot of guidance by more senior team members.

John Jarocki, john.jarocki@gmail.com

10.4. Online Analysis

Several sophisticated online analysis tools exist if an Internet connection is available to the analyst. Our team uses these sites to provide a quick triage, and in some cases, a comprehensive analysis of questionable web sites or captured malware. Orion contains bookmarks for submission of these samples to the following tools and more when online analysis is an option. (CWSandbox - Automated Malware Analysis, 2010), (jsunpack - a generic JavaScript unpacker, 2010), (ThreatExpert online file scanner, 2010), (VirusTotal - Free Online Virus and Malware Scan, 2010) (wepawet, 2010)

The following screen shot shows an analysis report from SunBelt Security's CWSandbox. This tool from Sun Belt Security executes the malware in a virtual environment and catalogs processes, files, network connections, etc. Even for experienced responders, this information can save a lot of time.

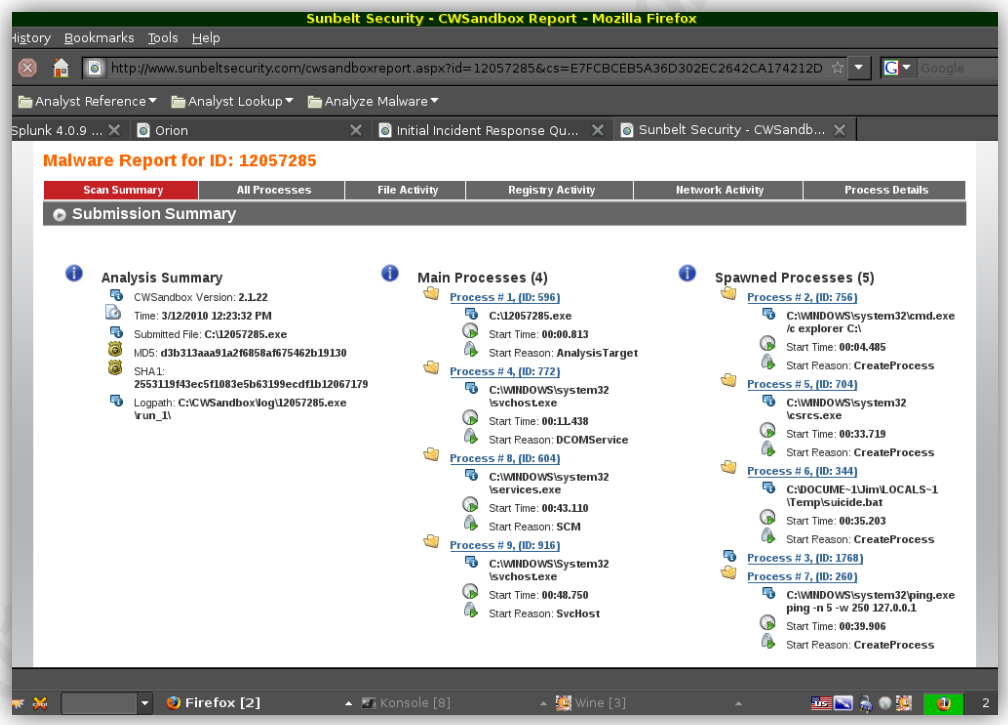

Figure 25: CW Sandbox analysis of captured malware

John Jarocki, john.jarocki@gmail.com

It's important to note that the owners of these sites often discourage automated submissions since these tools are often also used by attackers. For this reason, the analyst cannot rely on these automated tools alone.

11. Report Creation

Incident response and analysis work are not truly done until a report has been written. Orion provides report templates and a workflow to guide the responder to create the reports at the proper times. Here is a list of the various templates included currently in Orion:

- Incident Analysis Report
- Incident Activity Log
- Incident Responder Task List

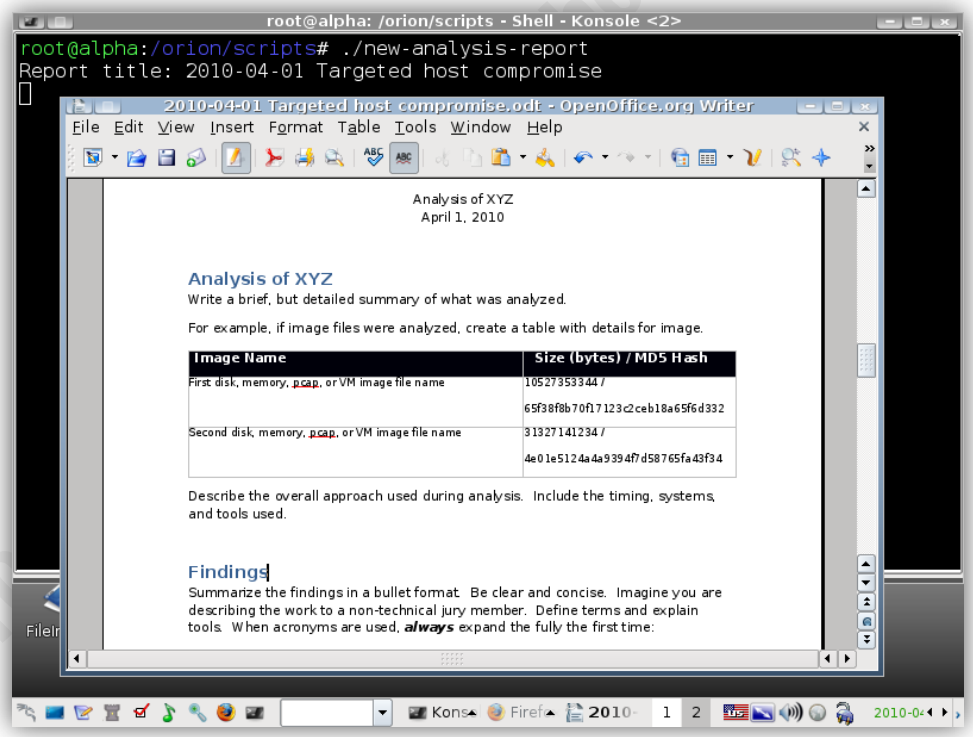

Figure 26: Analysis Report template

John Jarocki, john.jarocki@gmail.com

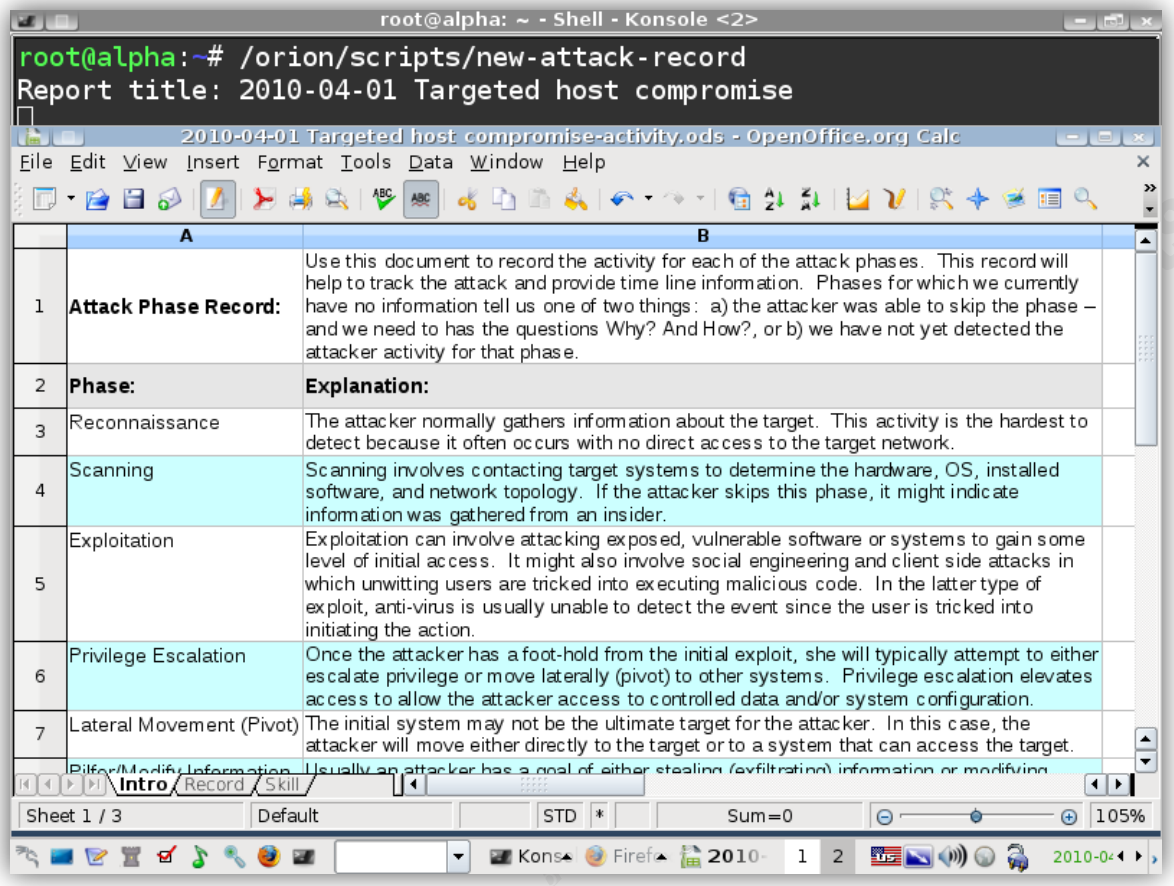

Figure 27: Attack Record template

The SANS Security Consensus Operational Readiness Evaluation (SCORE) web site has a number of forms created by students and instructors. The following are included with Orion:

- Incident Response Team Roster
- Incident Communication Plan
- Incident Task List
- Incident Containment
- Incident Eradication
- Chain of Custody Form

As with the other tools included in Orion, the report templates have been useful to our team in previous incidents and analysis. While these forms are not mandatory, they can

John Jarocki, john.jarocki@gmail.com

help instill a standard and repeatable incident response process. More importantly, they help responders get past the problem of starting from a blank page.

12. Availability of Orion

Orion is currently an internal project in use by the author and his colleagues. The design goals were written to satisfy the needs of our incident response team. Once that small team has reviewed Orion and provided input, the intent is to provide a public release. After it moves from "alpha" to "beta" release form, Orion will be made available at the Orion project web site (http://orionlivecd.sourceforge.net/). Requests for early copies or more information can be directed to orionlivecd@gmail.com.

13. System Requirements

Based on preliminary testing, the current system requirements for Orion are:

	CPU	RAM	Disk	Graphics
Minimum	900 MHz Celeron	512MB	0 (none required)	VESA
Recommended	2+ GHz Core 2 Duo	1GB	20 GB installed	NVIDIA GeForce 9400M, 256MB

Orion can be used as a Live CD, run from a USB token, installed directly on a hard drive, or used from within a virtual machine environment such as VMware. In the latter case, the software may again be used as a Live CD by booting any virtual machine with the orion.iso as the CD/DVD image or by installing Orion directly on the virtual machine hard drive as the operating system. The ubiquity installer is used to perform the hard drive installation, just as it is with BackTrack. Executing the install.sh in root home directory will start this process.

14. Tested Configurations

So far Orion has been tested using the following hardware and installation configurations.

Hardware	Installation Type	Recom-mended?	Notes
Dell D630	Live CD on bare hardware	No	Slow due to DVD access. Good graphics support.
Dell D630	Live CD in VMware from .iso	Yes	Live CD using ISO is much faster
Dell D630	Installed on hard drive	Yes	Fast. Good graphics support.
Dell E6400	Live CD boot direct on bare metal	No	Slow. Poor graphics support.
Dell E6400	Installed on hard drive	Yes	Fast, but poor graphics support.
Macbook Pro 13"	Live CD in VMWare direct from DVD	No	Heavy access of DVD
Macbook Pro 13"	Live CD from .iso in VMWare	Yes	

15. Future Work

Orion contains a significant number of scripts, tools, and incident response capabilities, it really only scratches the surface of the author's vision for it. There are many ideas that have not been implemented in the current version of Orion (Alpha). Examples of features being researched for future versions of Orion include:

- More sophisticated tunneling techniques such as Miredo (IPv6)
- More sophisticated awareness and containment capabilities, such as:
  - Deployment of honey tokens for detection
  - BotHunter or similar botnet detection tools
  - OSSEC to supplement local defenses
- Significantly more analysis and visualization tools
- Virtualization platform that can be redistributed
- Videos, tutorials, and other Quick Start documentation

John Jarocki, john.jarocki@gmail.com

16. Conclusion

BackTrack is a very successful Live CD for penetration testers. Although many information security professionals keep BackTrack in their toolkit, it does not provide the tools needed by security incident responders. Orion is an attempt to leverage the things that make BackTrack successful -- such as an easy to use and install distribution, a comfortable user interface, and an exemplary collection of tools -- and add the capabilities needed by incident response teams. Those include team communication, collaboration, and case tracking tools. Orion provides those as well as acquisition, awareness, and analysis software. Custom wrapper scripts tie these tools together. Finally, some report templates and even report writing tools are included to help the team report in a thorough and consistent manner.

17. References

Albert, C., Dorofee, A. J., Killcrece, G., & Zajicek, M. (2004). *Defining Incident Management Processes for CSIRTs: A Work in Progress.* Pittsburgh: Software Engineering Institute, Carnegie Mellon University.

Best Practical. (2010). *RTIR: RT for incident response*. Retrieved March 12, 2010, from Best Practical web site: http://bestpractical.com/rtir/

DEFT Linux - Computer Forensics Live CD. (2010). Retrieved March 10, 2010, from http://www.deftlinux.com/

e-fense, Inc. (2009). *Helix3 Pro 2009R2 User Manual.* Centennial: e-fense, Inc.

geek00l. (2010). *raWPacket*. Retrieved March 11, 2010, from HeX: http://www.rawpacket.org/projects/hex/

Gianluca, C., & De Franceschi, A. (2010, March 9). *Internet Traffic Decoder, Network Forensics Tool (NFAT)*. Retrieved April 17, 2010, from Xplico web site: http://www.xplico.org/

John Jarocki, john.jarocki@gmail.com

Henry, P. (2009, September 12). *Best Practices In Digital Evidence Collection*. Retrieved March 10, 2010, from SANS Computer Forensic Investigations and Incident Response Blog: http://blogs.sans.org/computer-forensics/2009/09/12/best-practices-in-digital-evidence-collection/

Khurana, H., Basney, J., Bakht, M., Freemon, M., Welch, V., & Butler, R. (2009). *Palantir: A Framework for Collaborative Incident Response.* New York: ACM.

Mandia, K., Prosise, C., & Pepe, M. (2003). *Incident Response and Computer Forensics, 2nd Ed.* Emeryville: McGraw-Hill Osborne Media.

Network Security Toolkit (NST v2.11.0). (2009, September 22). Retrieved March 10, 2010, from http://networksecuritytoolkit.org/nst/index.html

Offensive Security. (2010, April 19). *BackTrack Linux*. Retrieved April 19, 2010, from http://www.backtrack-linux.org/

Scarefone, K., Grance, T., & Masone, K. (2008). *Computer Security Incident Handling Guide, NIST SP 800-61-Rev1.* NIST.

Securix-NSM. (2010). Retrieved March 10, 2010, from SecurixLive: http://www.securixlive.com/knoppix-nsm/

Skoudis, E. (2009). Incident Handling Step-by-Step and Computer Crime Investigation. In *Security 504 Hacker Techniques, Exploits, and Incident Handling* (pp. 12-117). The SANS Institute.

Tshark examples: howto capture and dissect network traffic. (2009, December 13). Retrieved March 12, 2010, from CodeAlias: Networking & Coding Articles: http://www.codealias.info/technotes/the_tshark_capture_and_filter_example_page

John Jarocki, john.jarocki@gmail.com

18. Appendix A: Packages Added to Orion

anteater	etherape	grokevt	ollybone
argus	evans-debugger	gvim	OmegaT
arpalert	eventcombMT	hinfo	openoffice
arpwatch	exiftool	honeyd	pecarve.exe
bless	extractscripts.py	imsniff	pestat.exe
bvi	feagent.exe	ippl	planner
chaosreader	FileInsight.exe	ipsec-tools	pooltools
corkscrew	FindEvil.exe	knockd	rumint
citadel	FlexHex	lft	sancp
darkstat	ftp-imager	LockoutStatus.exe	tcpstat
denyhosts	fwanalog	logparser	tcpick
disktype	fwlogwatch	malzilla	volatility
dnstop	geoip-bin	notecase	xmount
esound	ggobi	OfficeCat	xplico
espeak	graphviz	OfficeVis	XORSearch.exe

19. Appendix B: BackTrack Tools Removed from Orion

airflood	exploitdb	sip-tester
airgraph-ng	fakeap	sip-tester-menu
airoscript	fuzzgrind	sipbomber
airpwn-ng	ghettotooth	spectools
airsnort	greenplaque	spectools-menu
ark-kde3	gstreamer0.10	spike-fuzzer
b43-fwcutter	hidattack	sqlbrute

John Jarocki, john.jarocki@gmail.com

battlife	hydra	sqlmap
beav	iaxflood	sqlninja
bed	jbrofuzz	sqlsus
beef	kamera-kde3	sshatter
bf2	mdk3	tbear
blackhat	medusa	tftp-bruteforcer
blindsql	medusa-menu	ussp-push
bluebugger	middler	ussp-push-menu
bluemaho	obex-data-server	valgrind
blueprint	obexftp	valgrind-menu
bluesmash	obexftp-menu	vncrack
bluesquirrel	obexstress	voiper
bluez-alsa	ohrwurm	w3af
bluez-gstreamer	openvas-client	waffit
brutessh	openvas-libnasl	wapiti
bsh	openvas-libraries	wapiti-menu
bsh-gcj	openvas-menu	webshag
bss	openvas-plugins	wepbuster
bunny	openvas-server	wfuzz
carwhisperer	pblind	wifizoo
cowpatty	powerfuzzer	wmat
dirbuster	pyrit	wsfuzzer
dkftpbench	rww-attack	xsss
dns-bruteforce	sip-rogue	zzuf

John Jarocki, john.jarocki@gmail.com