HW #6   
COMP5370/6370   
A. Skjellum, Instructor; A. Ravipati, TA   
Assigned October 28, 2015   
Due November 9, 2015 – 11:59pm

From the attached papers, give a write up of the following questions.

1. At the lowest levels of data gathering, what information did the investigators collect and what tool(s) did they use for that purpose?
   1. The aim of this present investigation is to further re ne the methodologies used to investigate and analyze malware networks through a *fusion methodology*, which combines network-based technical interrogation, data analysis and visualization, and eld-based contextual investigations (See Box 2, below). The combination of methods from different disciplines is a critical and common feature of both the *GhostNet* and *Shadow* investiga- tions and analyses. Network-based technical interrogation, open source data mining and analysis (using tools such as Google), key informant interviews and eld-based investigations on their own can accomplish a great deal, but it is through their fusion that a more comprehensive and nuanced understanding can be achieved.
2. How did they identify the malware (the Trojans) present in the infected computers?
   1. “After discovering several instances of malware on these servers, we set up a *honey pot* computer and were able to identify additional malicious servers by monitoring the traf c generated by our infected *honey pot*. Using the attacker(s)’ web-based administration interface, we were able to command our *honey pot* computer to download *gh0st RAT*, one of the Trojans used by *GhostNet*. Eventually, our *honey pot* computer established a connection to the attacker(s)’ *gh0st RAT* client. The attacker(s) proceeded to execute commands on our *honey pot*. We were able to discover several IP addresses within a DSL range in Hainan Island (PRC) that the attacker(s) used to communicate with computers infected with *gh0st RAT*.”
3. How did the investigators discover the identities (meaning the IP addresses and the geographic locations) of the computers used as the command and control centers? Also, what role was played by the computers used as control centers, and also, what role were the computers used as command centers?
   1. The investigation consisted of two distinct phases.   
      **Phase 1: Field-based investigations in India, Europe, and North America (June-November 2008)**Field research was carried out in Dharamsala, India, the location of the Tibetan Government-in-Exile. Follow-up research was conducted at Tibetan missions abroad in London, Brussels and New York. During this phase we had unprecedented access to the Tibetan government and other Tibetan organizations. This allowed us to establish a baseline understanding of information security practices at these locations and to design an evidence-based approach to the investigation.  
      We also conducted extensive on-site interviews with officials in the Tibetan Government-in-Exile, the private office of the Dalai Lama, and Tibetan non-governmental organizations. The interviews focused on the allegations of cyber espionage. We also sought alternative explanations for leakage of confidential documents and information and examined basic information security practices at these locations.  
      Network monitoring software was installed on various computers so as to collect forensic technical data from affected computer systems, and initial results were analyzed *in situ*.32 This initial analysis confirmed the existence of malware and the transfer of information between infected computers and a number of control servers.33  
      **Phase 2: Computer-based scouting, target selection, and data analysis (December 2008-March 2009)**During the second phase of the investigation, researchers based at the Citizen Lab analyzed the data collected by the field team.  
      The data collected in Dharamsala and at Tibetan missions abroad led to the discovery of four control servers and six command servers. These control servers were identified and geo-located from the captured traffic using a simple IP lookup.34 The control servers were then probed and web-based control interfaces were identified on four control servers, which allowed us to view and control the network. The system was actively monitored for two weeks, which allowed us to derive an extensive list of infected systems, and to also monitor the systems operator(s) as the operator(s) specifically instructed target computers.   
      The data collected during both phases was integrated in Palantir, a data visualization and analysis tool. The Palantir platform provides a data fusion and visualization environment that enhances analytical capabilities.
   2. Control servers, these interfaces allow attacker(s) to send instructions to, and receive data from compromised computers The control servers’ web interface contains three main components: 1) a listing of all the infected computers that have reported to the control server; 2) an interface to issue commands to the infected computers; and 3) an interface to monitor pending commands to infected computers and their results when completed. The commands issued to the infected computers direct the infected computer to download les from additional *command servers* under the attacker(s)’ control. In some cases, these servers act as control servers themselves; however, some appear to be used exclusively to host malicious les that infected computers are meant to download. The attacker(s) set commands on the control servers that instruct infected computers to download additional remote administration Trojans, such as *gh0st RAT*, in order to take complete real-time control of the infected computers.
4. What was the capability of the specific Trojan that played a large role in stealing information from the infected computers? How did this Trojan allow the humans to control the infected machines in real time?
   1. One of the commands available to the attacker(s) instructs infected computers to download the *gh0st RAT* remote administration tool, which gives the attacker(s) full, real-time control of the infected computer. *Gh0st RAT* is an open source Trojan that is widely available online. It was developed by Chinese programmers but has now been translated into English. The program allows an attacker to create an executable le that can be repacked and disguised and used to infect and compromise a target computer. This le can be con gured to directly connect to the *gh0st RAT* owner or to a third location, a control server, when it retrieves the current IP address of the *gh0st RAT* owner. (**See Fig. 10** - p. 36) Once the infected computer connects to the *gh0st RAT* owner, an entry appears in the *Connection* window with some information about the infected computer. The *gh0st RAT* owner may then issue commands to the infected computer. These commands include le manager, screen capture, keylogger, remote shell, system, webcam view, audio capture, as well as the ability to force the infected host to download and execute additional malware, such as a *gh0st RAT* update.
   2. The *GhostNet* system directs infected computers to download a Trojan known as *gh0st RAT* that allows attackers to gain complete, *real-time* control. These instances of *gh0st RAT* are consistently controlled from commercial Internet access accounts located on the island of Hainan, People’s Republic of China.
5. Look up articles that explain the Trojan’s components and capabilities and report on what you learn

**Gh0st RAT capabilities**Below is a list of Gh0st RAT capabilities. Gh0st RAT can:

* Take full control of the remote screen on the infected bot.
* Provide real time as well as offline keystroke logging.
* Provide live feed of webcam, microphone of infected host.
* Download remote binaries on the infected remote host.
* Take control of remote shutdown and reboot of host.
* Disable infected computer remote pointer and keyboard input.
* Enter into shell of remote infected host with full control.
* Provide a list of all the active processes.
* Clear all existing SSDT of all existing hooks.

**Gh0st RAT Components**This section will throw light on both at user and kernel level binaries of the Gh0st RAT toolset. Gh0st RAT has two main components: client and server.

Controller Application: This is known as client, which is typically a Windows application that is used to track and manage Gh0st servers on remote compromised hosts. The two main functions this module serves is the management and control of Gh0st servers and the ability to create customized server install programs.

Windows DLL (user level binary): The DLL is named SVCHOST.DLL. It is the Windows DLL that gets installed on a compromised host as a Windows service. This service is the server component of the Gh0st toolkit. It checks in to the Gh0st client on start-up and awaits instructions. The setup and installation of this DLL as a service is done by the install program (Dropper) SERVER.EXE which we will discuss in a short while.

INSTALL.EXE Dropper application is used to install SVCHOST.DLL. This is a stand-alone Windows application that contains all required code to prepare a compromised host for the installation of the Gh0st RAT server service and the launching of that service.

Kernel Level Binary: This is present in the toolset with the .SYS filename RESSDT.SYS. This is a very small device driver that performs a single task: resetting the Windows System Service Dispatch Table (SSDT). This is the only kernel level binary in the toolset. It runs at system startup on the compromised host and removes all hooks in the SSDT.

Install Program: This is commonly called “the dropper.” It contains the two above described binaries and performs all of the work necessary to install the Gh0st server on a host and startup the Gh0st service.

(<http://resources.infosecinstitute.com/gh0st-rat-complete-malware-analysis-part-1/>)

I learned that Gh0st RAT (Remote Access Terminal) is a trojan “Remote Access Tool” used on Windows platforms. It has been used to hack into some of the most sensitive computer networks on Earth. I learned about its components and its capabilities.

1. (Grad required, Ugrad extra credit). Download the Trojan into a linux machine (where it will do no harm) and see what you can infer about its capabilities. Report on what you find out and compare with what you learned in problem #5.