

The Command Structure of the Aurora Botnet

History, Patterns and Findings

Executive Overview

Following the public disclosures of electronic attacks launched against Google and several other businesses, subsequently referred to as “Operation Aurora”, Damballa conducted detailed analysis to confirm that existing customers were already protected and to ascertain the sophistication of the criminal operators behind the botnet.

There has been much media attention and speculation as to the nature of the attacks. Multiple publications have covered individual aspects of the threat – in particular detailed analysis of forensically recovered malware and explanations of the Advanced Persistent Threat (APT).

By contrast, Damballa has been able to compile an extensive timeline of the attack dating back to mid-2009 that identifies unique aspects to the Aurora botnet that have been previously unknown. Based upon this new information and our experience in dealing with thousands of enterprise-targeted botnets, Damballa believes that the criminal operators behind the attack are relatively unsophisticated compared other professional botnet operators. Even so, the results proved just as damaging as a sophisticated botnet since the threat was not quickly identified and neutralized.

Key observations discussed in the main body of this analysis report:

- The major pattern of attacks previously identified as occurring in mid-December 2009 targeting Google appear to originate in July 2009 from mainland China.
- Hosts compromised with Aurora botnet agents and rallied to the botnet Command-and-Control (CnC) channels were distributed across multiple countries before the public disclosure of Aurora, with the top five countries being the United States, China, Germany, Taiwan and the United Kingdom.
- Damballa identified additional botnet CnC domains used by these criminal operators and established a timeline of malware associations back to May 2nd 2009 by tracking the evolution of the malware used by Aurora’s operators.
- Analysis of network traffic associated with the lookups of the botnet CnC is not consistent with the publicly discussed Internet Explorer 6 infection vector.
- This botnet has a simple command topology and makes extensive use of Dynamic DNS (DDNS) CnC techniques. The construction of the botnet would be classed as “old-school”, and is rarely used by professional botnet criminal operators any more. Reliance upon DDNS CnC is typically associated with new and amateur botnet operators
- The criminals behind the Google attack appear to have built and managed a number of separate botnets and run a series of targeted attack campaigns in parallel. This conclusion is based upon CnC domain registration and management information. The earliest of the CnC domains associated with these botnets, reliant upon DDNS service provisioning, appear to have been registered on July 13th 2009.

- The botnet operators had access to large numbers of CnC hosts in geographically diverse hosting co-locations from the very start – a fairly high cost for a botnet. Further, the botnet employed over a dozen domains in diverse DDNS networks for CnC. Some of the botnet agents focused on victims outside of Google, suggesting that each domain might have been dedicated to a distinct class or vertical of victims.
- Only the US victims of the attack were compelled to perform mail-based DNS queries – an event that would typically indicate attempted document exfiltration via email services.
- Damballa identified multiple CnC testing, deployment, management and shutdown phases of the botnet CnC channels. Some of the CnC domains appear to have become dormant for a period of time after they infected victim systems. This type of activity can sometimes be associated with an update to the botnet malware, or when the criminal operator sells/trades a segment of the botnet to another criminal operator.
- The botnet operators behind the Aurora attacks deployed other malware families prior to the key Trojan.Hydraq release. Some of these releases overlapped with each other. Two additional families of malware (and their evolutionary variants) were identified as “Fake AV Alert / Scareware – Login Software 2009” and “Fake Microsoft Antispyware Service,” both of which employed fake antivirus infection messages to socially engineer victims into installing malicious botnet agents.

By studying the evolution of the Google attacks and tracking the malicious campaigns conducted before (and in parallel to) the public disclosure of “Operation Aurora” in January 2010, Damballa has established a detailed timeline of infections. Instead of this attack being a sophisticated APT operation, it appears that the attacks originated from a Chinese botnet operations team, and that the attack vector underwent several different phases of botnet building and malware deployment before being discovered by Google.

The fact that some of the later attacks utilized a different family of malware and may have exploited Zero-Day vulnerabilities within Internet Explorer 6 as one of the infection vectors is typical for modern botnet distribution campaigns. Botnet operators also increasingly trade or sell segments of the botnets they build. Once sold, the owner of the botnet typically deploys a new suite of malware onto compromised systems. The CnC provides the link between various campaigns run by the botnet operators and the multiple malware iterations. Since Damballa focuses on malicious, remote-controlled crimeware that depends on CnC to function, we were able to determine the evolution and sophistication of the Aurora botnet and its operators with greater detail and accuracy than other reports to-date.

In general, Aurora is “just another botnet” and typifies the advanced nature of the threat and the criminal ecosystem that supports it. It is important to note, however, that botnets linked to the criminal operators behind Aurora may have been sold or traded to other botnet operators, either in sections or on an individual victim basis. This kind of transaction is increasingly popular. Specialist botnet builders sell access to victim systems or networks for a fee – making it very simple for other entities to access confidential business systems and information without needing be technologically proficient. These transactions between criminals are very difficult to detect.

Introduction

The progression of semi-autonomous malware into globe-spanning botnets with victims numbering in the millions continues to accelerate. In short, botnets, and the criminal ecosystem that supports them, lie at the heart of modern cybercrime. Specialist contractors and service providers occupy every online niche, enabling both newbie hackers and professional botnet operators to overcome technological hurdles and operational barriers for a small price – typically stolen identities or access to hijacked systems rather than dollars.

All it takes to get started is an Internet search engine and the ability to install software on a computer. Devastating attacks start with a nominal fee for acquiring advanced malware construction tools capable of automatically generating customized botnet agents dramatically superior to tools used by professional hackers only three years ago. Fierce competition within the ecosystem has resulted in the commoditization of these tools and services, which has lowered price points and driven suppliers to differentiate with 24x7 support, money-back guarantees, replacement warranties and even SLAs.

Major international corporations have begun to publicly acknowledge this electronic threat. On January 12, 2010, Google announced that it had been the victim of a targeted attack and had subsequently identified over 34 additional organizations that had similarly been breached by the same criminal team. One major industrial powerhouse has publicly focused on the risks posed by persistent electronic attacks by including references to these threats in their quarterly 10-K filing.

Report Objectives

The purpose of this report is to explain the advanced state of today's botnet ecosystem by way of example, and to examine the ways in which criminal operators rely upon botnet technologies to breach corporate networks and extract secrets from their victims. Much media fervor has surrounded Google's public disclosure of the successful attacks against their systems. 33 other victims also fell prey to what has been frequently referred to as an Advanced Persistent Threat (APT). This report closely examines the methods employed by the criminal operators who conducted this botnet campaign.

Many security vendors have explained the operation against Google, dubbed "Operation Aurora," using a military vernacular. However, based upon analysis of exhaustive data surrounding these attacks and examination of both the malware and the CnC topologies used by the criminals behind Aurora, it appears that this threat can best be classified as a just another common botnet attack – and one that is more amateur than average.

This report details new analysis of the malware evolution and the CnC construction behind these attacks, and provides unique insight into similar threats facing large business. Comparisons are made between the Aurora attacks and professionally orchestrated campaigns run by sophisticated cyber criminals. Timelines track the evolution of this threat help to identify the objectives of the criminals behind the Aurora attacks, and illustrate the advanced state of the botnet ecosystem.

Understanding Aurora

Malware samples recovered from victim systems using forensic techniques lie at the heart of almost all public analysis of Aurora. The samples directly associated with Aurora are commonly referred to as Trojan.Hydraq. Damballa analyzed the Trojan.Hydraq outbreak using DNS monitoring logs obtained from CnC authority DNS servers. Since every infected host in the Aurora botnet contacted the

authority server, DNS logs provided a rich inventory of the botnet's resolution behavior. The logs also delivered insights into the development, gestation and growth of the Aurora botnet. This data leads to several interesting questions:

Origins – *Which network first resolved the botnet CnC domains? Who was the first victim? Are there clues in the first DNS lookups as to the authors or origin of the network?*

The analysis below shows that a university in China, and a Chinese collocation facility (colo), were critical early incubators of the infection. Portions of the infection originated from within Google China's offices.

Remediation and Damage Assessment – *Who else resolved the botnet CnC domains before news of the malware became public? What were the victim systems forced to do?*

Public accounts state that the botnet harvested email information. The DNS log analysis reveals numerous MX-lookups (mail-related DNS lookups). If these lookups are related to document theft, it is reasonable to estimate the number and timing of attempted exfiltration events. In addition to the type of DNS traffic, the log analysis also reveals where the victims are located.

Almost all (99%) of these events took place inside Google's US network. No victim in any other country performed MX lookups, suggesting Aurora's data exfiltration targets were all in the U.S. The pattern of MX lookups appears automated and lacks any diurnal properties.

Capabilities – *What else does DNS log analysis suggest, and what other questions does it raise about the attack?*

Damballa's analysis helps illumine the origin of the botnet, based on years of observing the authority servers used in the Aurora CnC.

Previously Disclosed Aurora Knowledge

"Operation Aurora" refers to the investigations of a cyber attack which appeared to have begun in mid-December 2009 and continued through to February 2010. Aurora was first publicly disclosed by Google on January 12, 2010 – and is commonly associated with attacks originating from China. The Aurora name was originally publicized by Dmitri Alperovitch, Vice President of Threat Research at McAfee, and refers to a file path artifact that might reveal what the criminal authors of the malware named their operation.

Key facts publicly associated with Aurora:

- a) Google stated that some of their intellectual property had been stolen and publicly announced the attack on January 12th 2010.
- b) While the scope of reported victims includes around 34 organizations, only Google, Adobe Systems, Juniper Networks and Rackspace have publicly confirmed that they were targeted. Various media reports have stated that Yahoo, Symantec, Northrop Grumman, Dow Chemical and the Rand Corporation were also among the targets.
- c) Many security agencies and experts claim the attack to be a sophisticated use of "advanced" tools and techniques – most notably the use of a Zero-Day exploit for a previously unknown vulnerability in Microsoft's Internet Explorer 6 browser technology.
- d) The public name for the malware component that allowed the Aurora criminal operators to remotely control their victims system is called Trojan.Hydraq.
- e) The Aurora attacks are widely assumed to be an APT originating from within China.

Advanced Persistent Threats

Advanced Persistent Threats (APTs) are a cybercrime category directed at business and political targets. APTs require a high degree of stealthiness over a prolonged duration of operation in order to be successful. The attack objectives therefore typically extend beyond immediate financial gain, and compromised systems continue to be of service even after key systems have been breached and initial goals reached.

Definitions of precisely what an APT is can vary widely, but can best be summarized by their named requirements:

Advanced – Criminal operators behind the threat utilize the full spectrum of computer intrusion technologies and techniques. While individual components of the attack may not be classed as particularly “advanced” (e.g. malware components generated from commonly available DIY construction kits, or the use of easily procured exploit materials), their operators typically access and develop more advanced tools as required. They combine multiple attack methodologies and tools in order to reach and compromise their target.

Persistent – Criminal operators give priority to a specific task, rather than opportunistically seeking immediate financial gain. This distinction implies that the attackers are guided by external entities. The attack is conducted through continuous monitoring and interaction of a botnet in order to achieve the defined objectives. It does not mean a barrage of constant attacks and malware updates. In fact, a “low-and-slow” approach is usually more successful.

Threat – Means that there is a level of coordinated human involvement in the attack. The criminal operators have a specific objective and are skilled, motivated, organized and well funded.

Damballa’s Perspective

Damballa’s research and technical expertise focuses on the detection of CnC tethering and the malicious communications between a victim’s computer and the remote criminal operator. Damballa detects new botnet CnC channels as they are created and used by criminal operators. This globe-spanning array of network sensors monitors CnC use to identify victims that join botnets.

Damballa used key DNS observations about the operational characteristics of Dynamic DNS zones (e.g. zone cuts, TTL changes, NS changes, etc.) in order to identify the different states in which the botnet was operated by its criminal controllers. Changes in the way that a DNS zone is structured by criminals typically denotes an intention to develop, test, and operate malicious infrastructure, or abandon a particular zone and move to a new one. Damballa also reviewed historical DNS resolution data derived from our passive observation systems to identify when (and how frequently) the CnC domain names associated with the Aurora botnet were queried. This information provided valuable insight into the pace at which victims rallied to the botnet and established a timeline for Aurora.

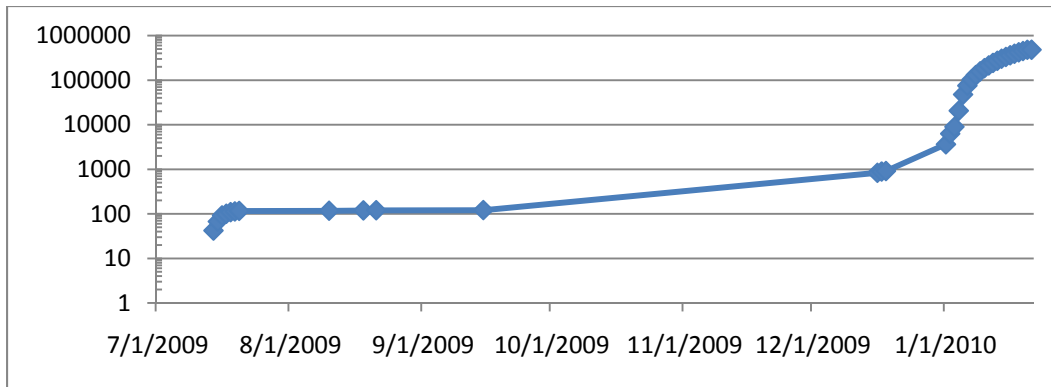


Figure 1: Cumulative volume of CnC domain name resolutions. Absolute numbers do not represent individual victims (i.e. victim computers make repeated lookups based upon the TTL of the CnC domain and relative malware activity on the system), but do depict approximately when the CnC domains were first used by the Aurora botnet. From this passive DNS resolution dataset, that date appears to be June 14, 2009.

These network observations combine with Damballa’s ability to identify Zero-Day remote access malware and botnet agents within customers’ networks to determine additional CnC relationships. Zero-Day malware samples are automatically passed to Damballa’s analysis cloud – along with tens-of-thousands of new malware variants obtained through industry security sharing programs. These network behaviors are extracted, and provide Damballa with additional insight into CnC evolution and criminal ownership. They also allow us to cluster various malware and botnet agents automatically with their respective criminal operators – despite factors such as serial variant production, migrations to new malware families and sub-contracting malware development to other criminal authors.

Trojan.Hydraq is the name of a family of malware now synonymous with Operation Aurora. To date, only a handful of related samples have been made public by various security vendors – almost all of which were gathered through forensic analysis of compromised computers. However, it is important to understand that not only are there multiple variants of malware within the Trojan.Hydraq family, but that criminal operators also use(d) other malware families in their attacks. Based upon analysis of samples and data gathered by Damballa, malware associated with the criminal operators behind the Aurora botnet can be traced back to August 2009.

A holistic DNS forensic analysis of any botnet that utilizes DNS as a critical communication element requires DNS information from both the iterative and recursive DNS phases. Utilizing large scale passive DNS information from large ISPs and DNS traces from a significant portion of the CnC’s DNS authority servers (ANS) Damballa has identified more than many infected hosts that attempted to connect or rally to the five CnC domain names associated with the Aurora botnet and investigated in this report. These hosts were distributed across multiple countries at the time of the public Google disclosure (January 12, 2010).

Position	Country
1	United States
2	China
3	Germany
4	Taiwan
5	United Kingdom

Table 1: Top 5 countries with Aurora botnet victims

Damballa’s passive DNS data collection indicates that the infection vector was not centralized, and that a significant number of infected assets tried to look up CnC domain names throughout the US, with a higher frequency in the Northeast.

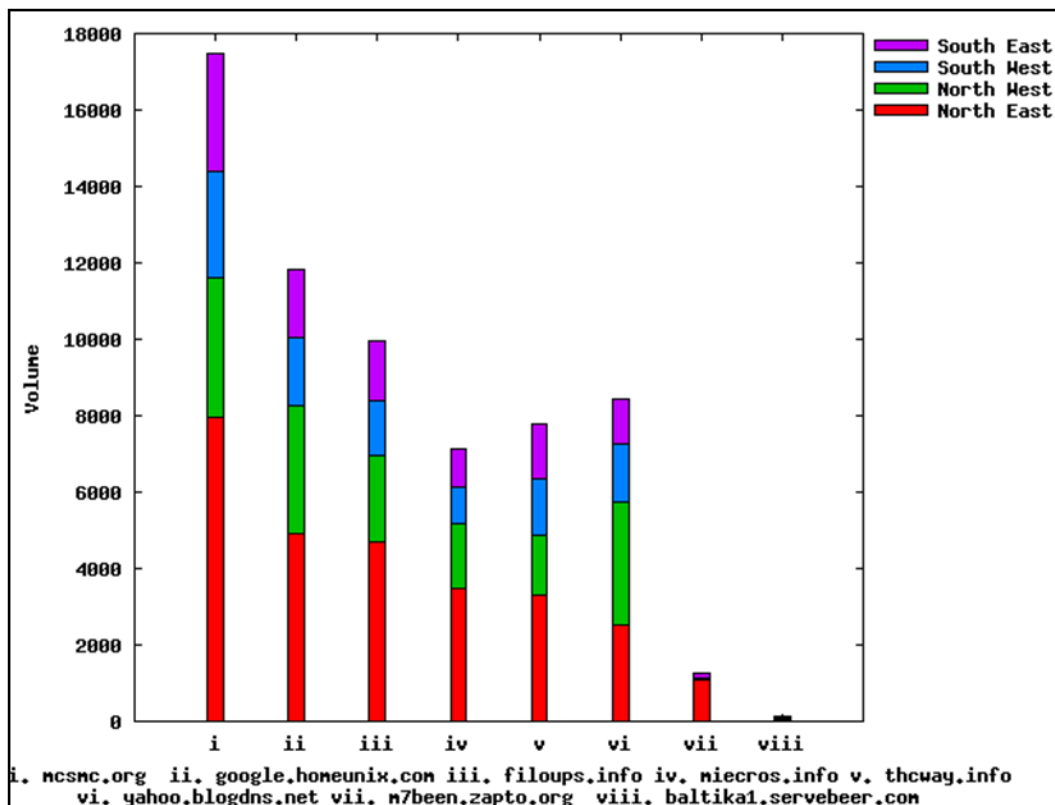


Figure 2: Volume of DNS queries per Aurora CnC domain associated with the attacks within the USA, by geographic region

Some interesting observations can be made about the lifetime and “popularity” of the CnC domains used. The next figure shows that portions of the CnC domain names were active since the beginning of September 2009 (e.g. google.homelinux.com, yahoo.blogdns.net, mcsmc.org). These domain names reveal two important trends – a downward-spike during the month of October and a steady hit rate for the remaining months. Beside these long-lived CnC domain names, Damballa observed a number of domain names that become active in the early days of November. Some of them were active only for a couple of months (e.g. filoups.info), while others were active longer

(e.g. `m7been.zapto.org`, `baltika1.servebeer.com`, etc.) before they were sinkholed by corresponding DNS operators.

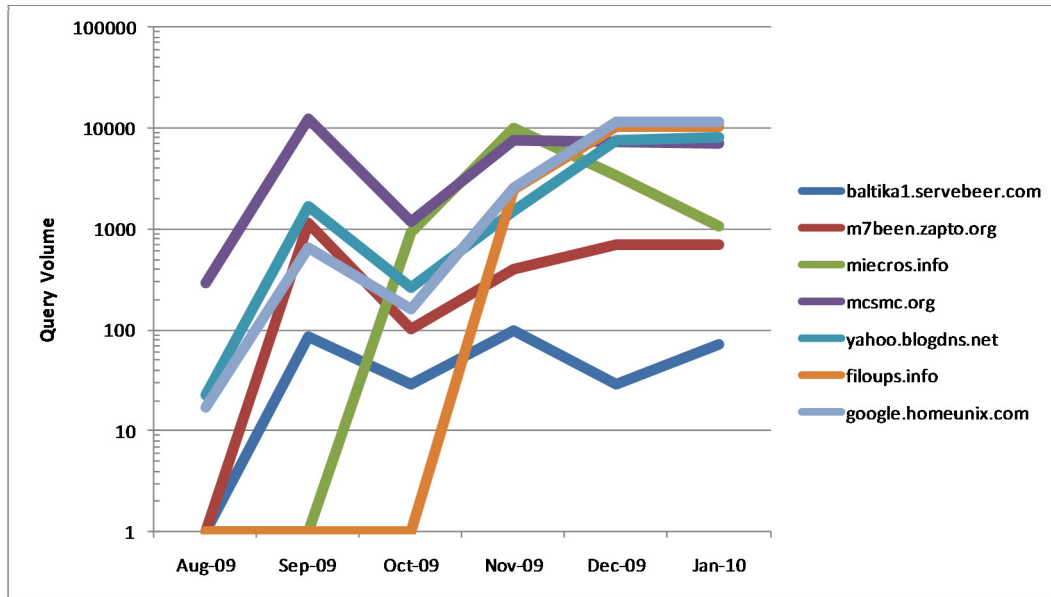


Figure 3: Volume of DNS resolution queries per Aurora botnet, per month. Spikes in query volume typically indicate growth of a botnet and renewed CnC interaction.

The Major Components

Botnets are a business. Professional criminal operators employ specialist tools, services and methodologies to conduct their botnet operations. While botnet discussion has been tied malware families in the past (e.g. “The Conficker Botnet”, “The Koobface Botnet”), today’s botnet operators regularly employ multiple families of malware, considering them disposable attack tools. The key elements of a botnet are:

Malware – The tool used by botnet operators to conduct malicious activities on victims’ computers and to provide remote control capabilities.

CnC – The electronic tether between the criminal operator, a control server and victims’ computers.

CnC Domain – The domain name of the host being used for CnC conduct or to route communications between the control server and the victim’s computer.

CnC Server – The server used by the botnet operators to rally and provide electronic tethers to victim computers.

Botnet – The collective name for malware-infected victims with established connections to a CnC server and remotely controlled by criminal operators.

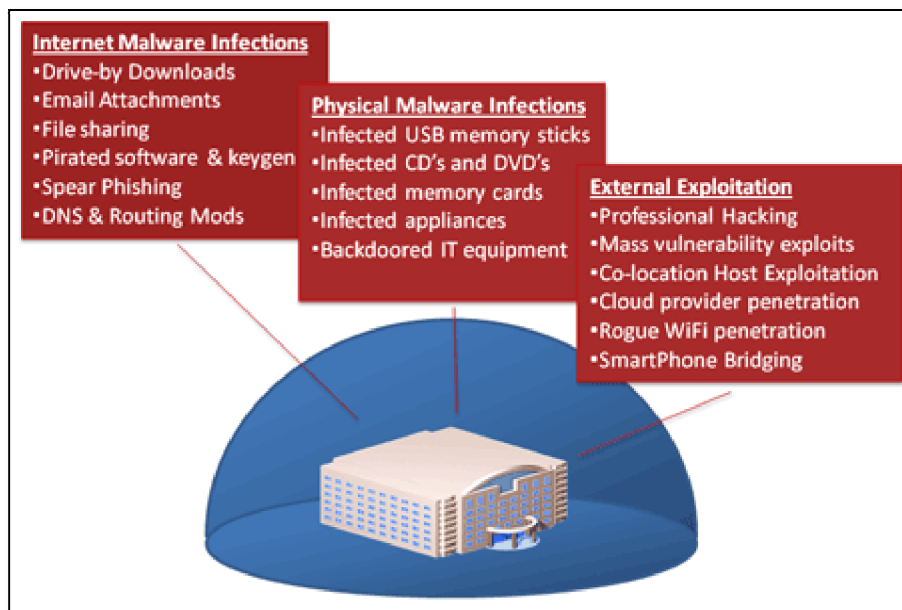
Criminal Operators – The person or team that builds, manages and reaps financial reward from a botnet.

How Advanced Persistent Threats Breach Enterprises

APTs breach enterprises through a wide variety of vectors, even in the presence of properly designed and maintained defense-in-depth strategies:

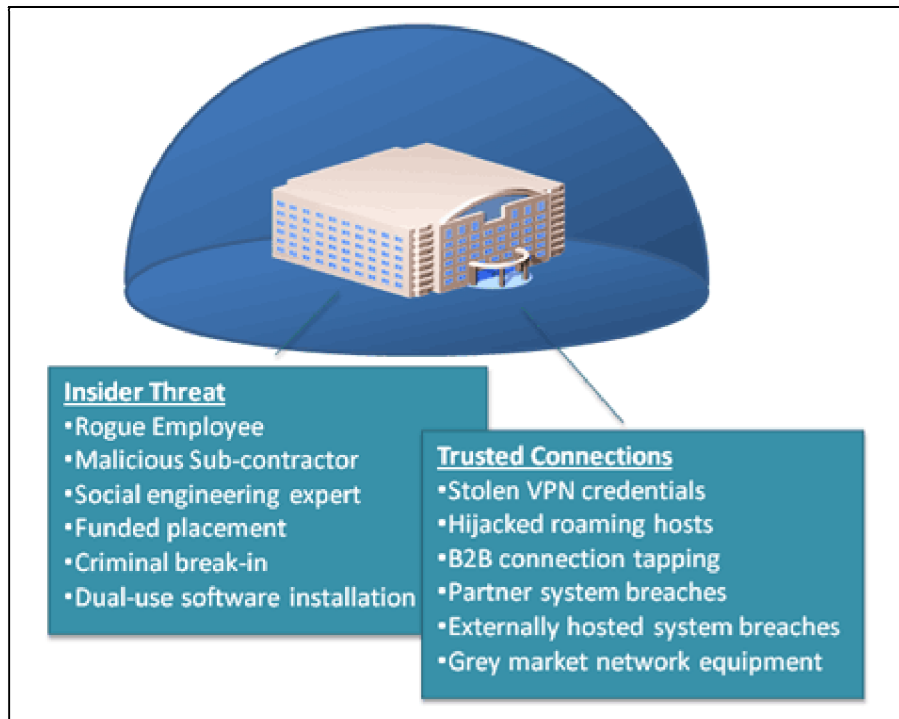
- Internet-based malware infection
- Physical malware infection
- External exploitation

Well funded APT adversaries do not necessarily need to breach perimeter security controls from an external perspective. They can, and often do, leverage “insider threat” and “trusted connection” vectors to access and compromise targeted systems.



Abuse and compromise of “trusted connections” is a key ingredient for many APTs. While the targeted organization may employ sophisticated technologies in order to prevent infection and compromise of their digital systems, criminal operators often tunnel into an organization using the hijacked credentials of employees or business partners, or via less-secured remote offices. As such, almost any organization or remote site may fall victim to an APT and be utilized as a soft entry or information harvesting point.

A key requirement for APTs (as opposed to an “everyday” botnet) is to remain invisible for as long as possible. As such, the criminal operators of APT technologies tend to focus on “low and slow” attacks – stealthily moving from one compromised host to the next, without generating regular or predictable network traffic – to hunt for specific data or system objectives. Tremendous effort is invested to ensure that malicious actions cannot be observed by legitimate operators of these systems.



Malware is a key ingredient in successful APT operations. Modern “off-the-shelf” and commercial malware includes all of the features and functionality necessary to infect digital systems, hide from host-based detection systems, navigate networks, capture and extricate key data, provide video surveillance and deliver silent covert channels for remote control. APT operators often use custom malware tools to achieve specific objectives and harvest information from non-standard systems.

At the very heart of every APT lies remote control functionality. Criminal operators need this capability in order to navigate to specific hosts within target organizations, exploit and manipulate local systems, and gain continuous access to critical information. If an APT cannot connect with its criminal operators, then it cannot transmit any intelligence it may have captured. In effect, it has been neutered. This characteristic makes APTs appear as a sub-category of botnets.

While APT malware can remain stealthy at the host level, the network activity associated with remote control is more easily identified. As such, APTs are most effectively identified, contained and disrupted at the network level.

Controlling the Victim

Once the victim’s computer has been compromised, the malware component will typically establish its first CnC session to register itself with the botnet CnC server. In order for this to occur, the botnet operator must correctly set up the CnC servers and also configure appropriate resolution services such as registering domain names and configuring DNS resolution settings.

Depending upon the sophistication of the botnet operators, this CnC infrastructure can take on many different forms, with each alternative offering varying degrees of robustness and flexibility. Readers are encouraged to read Damballa’s earlier whitepaper titled, “Botnet Communication Topologies: Understanding the Intricacies of Botnet Command-and-Control,” for more information on this topic.

Detailed analysis of DNS intricacies for CnC domain name querying and management follow.

Key Concepts: DNS Overview

DNS resolution can be generally viewed as having two phases – a private stub (or “recursive”) layer, and a public authoritative (or “iterative”) layer.

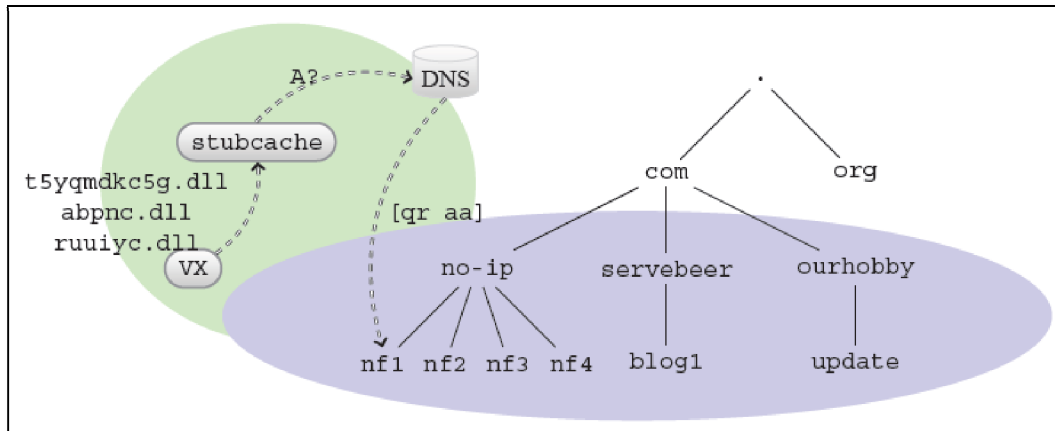


Figure 4: Conceptual view of Aurora DNS lookups and multiple monitoring opportunities. Damballa used the convenience of an authority monitoring system to gather [qr aa] responses.

The figure above illustrates how Aurora victims performed DNS lookups, and provides a simplified delegation tree for several of the Aurora-related CnC domains. An Aurora authority DNS zone is depicted: the light blue zone delegated to No-IP. The No-IP zone has been simplified in the diagram to include the authority DNS servers, `nf [1 - 4] .no-ip.com`, as well as the actual Aurora CnC domain, `blog1.servebeer.com`, even though in practice these are separate delegations from the `.com` TLD parent. An infected host is depicted in the light green area. Its resolution path consists of the virus code (designated as VX), a local stub resolver (often available through various statically named or random DLL files on Windows hosts), and a local recursive DNS server. The “private” portions of DNS traffic occur within this local envelope, colored as the light green area. No DNS monitoring takes place here, in part because of the possible presence of PII, and because of the staggering volume of traffic monitoring might entail, for even a small network. Such networks often generate billions of queries per day below the recursive.

When a victim attempts to contact the Aurora CnC domain `blog1.servebeer.com`, it must first discover the delegation of the zone to the No-IP authority name servers. (To save space, these steps are not shown in the figure above). The overall delegation of authority is shown in the figure as a tree. The hosts `nf [1 - 4] .no-ip.com` are the authorities for the CnC zone. Thus, the victim network’s recursive server discovers these name servers, queries for the Aurora CnC domain, and caches the answer.

Dynamic DNS and IP-Agility

Botnets have used Dynamic DNS services (DDNS) for nearly 8 years. For the most part, the role of DDNS in professional, criminal botnets is historic. Concentrated cleanup efforts and a few well documented arrests have changed the class of botmaster using DDNS. For the most part, professional cyber criminals do not use DDNS for botnet rallying, since DDNS providers:

- a) are generally responsive to law enforcement;
- b) keep logs; and
- c) a few are famously known to actively monitor and remediate their networks.

Since 2007, most “professional criminal” botnet CnCs (e.g., Russian mafia) have moved away from DDNS, because of the aggressive stance taken by the major DDNS providers against botnet abuse. While there has been a recent return of novice botmasters to the free DNS services, the professional criminal botmasters have largely moved on to more resilient, agile DNS technologies. For example, professional botnets buy tens of thousands of domain names, and use domain agility instead of the IP agility found in DDNS. The best example of this is Conficker.C. The decline in “professional” botnet use of DDNS services has been so dramatic that many anti-botnet researchers changed their focus to new areas of threat.

The average botmaster still using DDNS is generally a novice, and the malware they seed on victim machines is often kit-generated. There are a few exceptions where amateur botmasters evolve into professionals, but the bulk of botnets relying upon DDNS remain novice efforts, and use only a few domain names with a single DDNS provider.

The Aurora botnet uses DDNS and “old school” coordination techniques not used by sophisticated botmasters who have the means to purchase and manage dozens of domain names. And yet despite having the signature of a novice effort, it also used numerous different DDNS services.

Network Analysis

The network analysis in this report encompasses the CnC domain names known to be publicly associated with the Aurora attacks, plus an additional four non-public domains (listed below) which are related to the criminal operators behind the Aurora attacks through shared DDNS registration credentials and their synchronized management.

Domain	Authority Creation Date (UTC)
CnC_Domain.1	December 15, 2009
CnC_Domain.2	December 15, 2009
CnC_Domain.3	July 13, 2009
CnC_Domain.4	December 15, 2009
blog1.servebeer.com	December 15, 2009

Table 2: DDNS botnets with characteristics identical to the Aurora botnet and shared DDNS credentials. The first four of these CnC domains have been intentionally obscured.

The DNS TTL data associated with these interlinked Aurora domain names reveals that there were different phases to their use. The figure below indicates when a particular CnC domain name was sinkholed or idle (i.e. not pointing to a specific Internet IP address, or pointing to a local loopback address such as 127.0.0.1), it was pointing at probable development IP addresses as the criminal operators experimented with their attack tools, such as when the CnC domain names were pointing at the IP addresses associated with two of the CnC servers used during the Aurora attack.

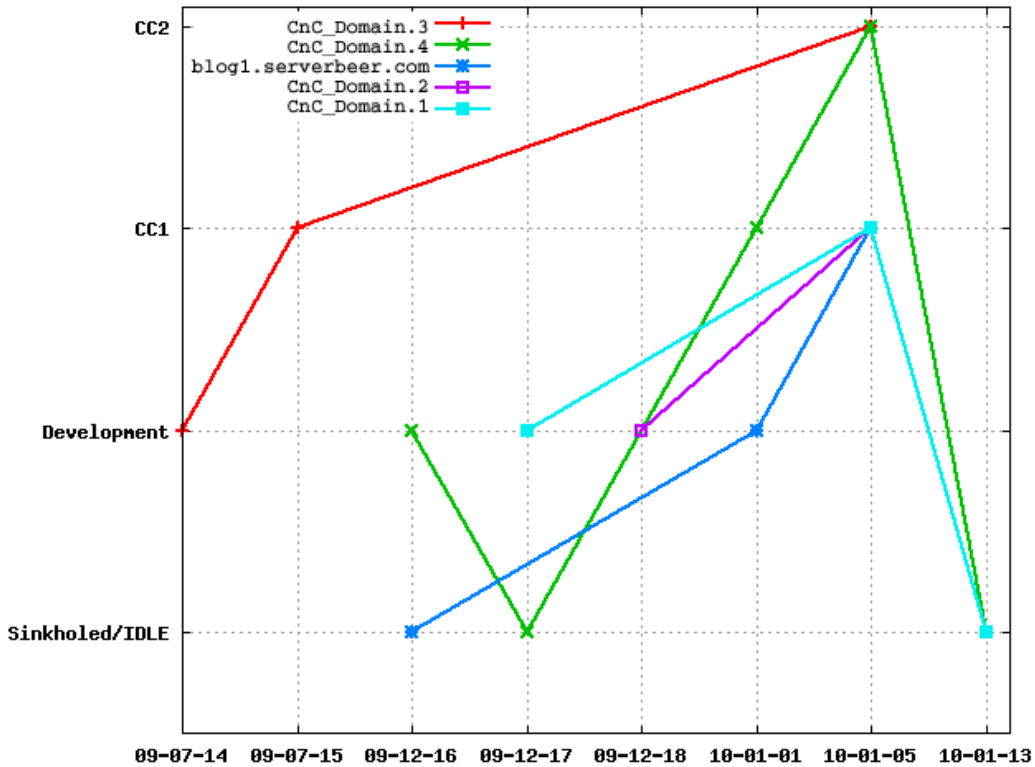


Figure 5: CnC domain name transition changes as the attackers developed botnet attacks.

Based upon passively obtained DNS resolution data from sensors scattered around the globe (but predominantly US based), Damballa observed that several key CnC domains resolved to different server IP addresses over the period of study. The transitions from one IP address to another can be used to identify the different phases of botnet development (e.g. as depicted in the figure above), as well as the nature of the CnC servers hosting and botnet topology (e.g. whether parts of the CnC network were using fast-flux services). The table below lists the number of IP address changes to the CnC domain name resolution – and is a lower bound number, since Damballa does not monitor all Internet traffic.

CnC Domain Name	Distinct IP Addresses
baltika1.servebeer.com	50
m7been.zapto.org	50
miecross.info	4
mcsmc.org	3
yahoo.blogdns.net	5
filoups.info	2
google.homeunix.com	2

Table 3: The number of distinct IP addresses observed by Damballa and associated with each of the CnC domain names for the period of August 2009 to the Google Aurora disclosure on January 12, 2010.

Overview of CnC Domains

Not all of the authority servers hosted by the DDNS providers for this botnet were monitored by Damballa and sampling practices were adopted for this analysis. In general, for large botnets, the sampling this produces is more than adequate to detect “professional cyber criminal botnets.”

Around 5,236 recursive DNS servers visiting the Aurora CnC authorities used BIND. The table below lists the major types. Damballa identified a signature specific to Chinese closed recursive DNS servers that provides policy insight to some selected resolvers.

The table below provides counts of queries from recursive DNS servers for both ISO-3166 country code and qtype. All data was gathered on or before January 11, 2010 (the eve of the Google public announcement) to avoid polluting queries from the press and researchers. It is estimated that Google discovered this attack in mid to late December, 2009, so some of the resolution traffic could be associated with their internal remediation.

The table also demonstrates that only US victims were required to perform MX queries, hinting at data extraction via SMTP mail services.

Query Type	US	CN	Others
15 (MX)	143,015	0	0
1 (A)	52,787	644	676
28 (AAAA)	12,254	84	0

Table 4: Breakdown of qtype by country code of recursive, for all five studied Aurora botnet CnCs. Highlights: (a) Only the US victims were compelled to perform MX queries (qtype 15); all networks in China and other countries never performed an MX query; (b) No AAAA (qtype 28) queries were performed by international victims, who were presumably collateral victims; the pairing of AAAA to A queries is discussed below; and (c) Most queries were MX (68% overall), and the attack heavily biased towards the US (also 68 % overall).

CnC Domains over Time

Damballa’s analysis of DNS data has revealed the very early origin (July 2009) of the botnet. Even during this early deployment, the botnet was widely dispersed. Since these were the first DNS resolutions for these attacks, it is reasonable to assume they are associated with the botmaster (e.g., testing or configuring their attack), and not victims. Thus, these resolutions might correspond to several CnC sites. If this theory is correct, it suggests that, despite using “naive” DDNS services typical of novice botnet operators, the Aurora botmasters had considerable resources available to them.

CnC_Domain.1

The first resolution for CnC_Domain.1 came from within Google China’s offices. It was followed hours later by resolutions inside Google’s offices in Mountain View, California. The pattern of lookups is remarkable, and is worth closer study. The first queries for CnC_Domain.1 were:

- 2009-12-16 05:26:44 AAAA (Google China)
- 2009-12-17 22:39:09 AAAA (Google Mountain View)
- 2009-12-17 22:39:09 A (Google Mountain View)

Counting Attempted Exfiltration Events

Other patterns of DNS messages in `CnC_Domain.1` suggest the attempted exfiltration of data.

Consider this ordering of queries:

2009-12-18 06:29:09	MX	(Google Mountain View)
2009-12-18 06:29:09	A	(Google Mountain View)

The queries both happen in under a second, indicating that a host using a recursive resolver wished to send email to the `CnC_Domain.1` CnC (hence the MX lookup). Dynamic DNS zones, however, almost *never* have valid MX RRsets, or if they do they are pointed to `abuse@traps` or `spamtraps`. Only a few DDNS providers offer mail, and the first query was therefore answered with an empty record (NOERROR status, with zero answers). As a result, the victim immediately performed an A query, to use the IP address for email. Whether these queries were followed by actual or successful email events is not known.

All MX queries in `CnC_Domain.1` came from the United States (and *no* other network outside the US performed such a query before the news broke). Before January 10th 2010, some 110,810 MX queries came from Google Mountain View, and one came from Comcast (San Jose). This Comcast-based query may have been testing by a Google security engineer, or it may have been an infection on a notebook after work (since the query took place in the late evening hours, PST).

From the volume of messages, it is presumed that each MX query corresponds to a single email exfiltration attempt. It would be hard to imagine a botmaster being able to direct these events individually. Thus, it may not be the case that bots were instructed to email materials when a specific event took place. Or the consistent pattern of queries could be the result of persistent searches of a hard drive, and attempted, periodic exfiltration of any useful data. This conclusion is speculation.

The lack of any diurnal pattern to these events does indicate that the trigger event for an MX lookup was not human-driven (e.g., the arrival of email on a victim machine, or selected actions by the botmaster). It is not known what information was taken, if any, or if these queries were in fact victim behavior. Public accounts from Google indicate that the attackers sought email records of human rights activists.

It is speculated that Google would have prevented the “direct-to-MX” behavior of hosts within their network. That is, in many corporate networks, individual user machines are prohibited from sending email directly, and must instead use a smart host or authenticated relay system. Thus, these MX lookups may well be side effect of an unsuccessful exfiltration effort. The malware also used ports 443 and 8585 for CnC, and could be instructed to perform any command.

CnC_Domain.3

The `CnC_Domain.3` CnC domain is interesting because of its age. The botnet dates back to July 14, 2009, fell dormant for months, and then became active again within Google’s network. Of the five CnC domains studied in detail within this report, this is the oldest, and most strongly suggests an origin for the botnet.

The early queries for the Aurora CnC domain `CnC_Domain.3` took place in the HangZhou region, with some occurring in Beijing. The domain had a remarkable number of queries from mainland China

and collocation facilities in the US within minutes of being created. Seconds later, another query came from Chinanet's network in the Chongqing area. The close timing of these suggests the owners of `CnC_Domain.3` had access to ISP, university, and commercial transit.

2009-07-14 02:50:03	A	(HiNet Taiwan)
2009-07-14 02:57:38	A	(CHINANET Jiangsu)
2009-07-14 02:58:31	A	(CHINANET HangZhou)
2009-07-14 03:03:11	A	(HangZhou Institute of Electronic Engineering)
2009-07-14 03:03:44	A	(CHINANET Chongqing)
2009-07-14 03:04:28	A	(FDC Servers, US Chicago)
2009-07-14 03:13:18	A	(Level 3, US Washington)

The pattern of these lookups suggests that the author was performing testing, and had access to two different transits (e.g., a school network and an ISP).

CnC_Domain.2

The first query for the `CnC_Domain.2` domain came from Google's Mountain View recursive.

2009-12-17 22:39:09	AAAA	(Google Mountain View)
2009-12-18 06:27:58	MX	(Google Mountain View)
2009-12-18 06:27:58	A	(Google Mountain View)
2009-12-18 18:15:18	AAAA	(Comcast; San Jose)
2009-12-18 18:15:18	A	(Comcast; San Jose)
2009-12-18 18:15:18	MX	(Comcast; San Jose)
2009-12-18 18:19:30	AAAA	(Google-IT)
2009-12-18 18:19:30	A	(Google-IT)

The `CnC_Domain.2` CnC domain is also notable because it witnessed queries from many other networks outside of Google before the public news broke. This domain has never been identified publicly as part of Aurora. Networks performing queries up to January 10, 2010 include numerous ISPs.

Observed Loss of Queries

When a botnet is remediated at the DNS level, the associated victims continue to query the authority DNS server. Unless and until the local network cleans the hosts or imposes network blocks, victim traffic to the authority will continue. A sudden loss of network traffic from a country, however, can be unusual, particularly where the victims are spread over disparate (heterogeneous policy) networks.

That is, it is unlikely that many different networks would *simultaneously* remediate hosts. Thus, while it may seem likely that all victims in a single network disappear (e.g., as when a network operator deploys a firewall rule), it is remarkable when all victims in diverse policy boundaries also disappear. Such centralized control speaks to the management of the botnet, and gives clues as to the policy preferences of the botmaster to attack/not attack a given suite of networks or countries.

Hosts performing DNS queries exhibited a random pattern of A queries. The TTL periods for the CnC domains was always short, meaning there was only a short period of time during which a stub query could be answered from cache, and not recorded at the authority. This behavior is typical of fast flux

networks. An increase in TTL from 60 to 360 seconds was identified, which signifies the cut over from the default zone TTL to the `SOA.minimum` used for wildcarded domains. Thus, the DDNS domains used in the attack appear to have been deregistered before December 18 and remained “open” for anyone in the world to register until the first week of January 2010.

The Malware Evolution

Aurora malware families date as far back as August of 2009. This trail helps determine the evolution and common characteristics of malware used by Operation Aurora, as well as a common *modus operandi* on the bot agents deployed as part of the attacks. The result is more than just an analysis of individual malware families. Rather, it helps profile the criminal operators behind Aurora via:

Malware Delivery Method – How does the malware get into the system? Is there a common delivery method or is it random?

System Behavior – Are the symptoms evident in the system common to all Aurora malware families or do they differ? Do the families use the same infection techniques, protection mechanisms and/or AV evasion techniques?

Network Behavior –Do the malware families exhibit the same network behavior?

CnC Server Trials Powered by Zero-Day Malware Variants

The table below lists significant events in the deployment and use of one of the Aurora botnet CnC servers known to the public, `filoups.info`, based on our data mining and analysis of malware samples and network traffic collected by Damballa. Several initial trials were conducted by the botnet operator prior to the “production” use of this CnC server.

The first set of CnC domains appeared in a FakeAV Trojan malware family1-a in the beginning of May 2009. There were several variants of the family1 malware in the wild in 2009. The second set of CnC domains was used by a new family1-b malware variant in October 2009. By leveraging new Zero-Day malware variants, the botnet operator(s) could easily evade AV product detection and experiment with different CnC domain construction and communication. For example, different combinations of CnC domains were tried by both family1-c and family1-d malware variants in late October 2009. Finally, the CnC domain `filoups.info` was deployed and used by malware family1-e in November 2009.

Domain	<code>mcsmc.org</code>	X	X	X	X	X
	<code>thcway.info</code>	X	X			
	<code>miecros.info</code>			X		
	<code>mnprfix.cn</code>				X	
	<code>micronetsys.org</code>				X	
	<code>filoups.info</code>					X
MD5		family1-a	family1-b	family1-c	family1-d	family1-e
Date		5/2/2009	8/18/2009	10/20/2009	10/22/2009	11/26/2009

Table 5: Botnet CnC trial evolution powered by Zero-Day malware variants.

The family1-e malware is part of Fake AV Alert/Scareware family analyzed below. The behavior of Fake Alert/Scareware is quite similar to Trojan.Hydraq malware associated with the actual Aurora attacks, albeit in a much more primitive form.

Sample Analysis Details

The additional samples in Damballa's possession that have been clustered as part of Aurora botnet malware can be separated into two distinct families of Fake AV Alert / Scareware: Login Software 2009 and Microsoft Antispyware Services. The first samples of each family were discovered by Damballa on November 26 2009 and August 19, 2009 respectively. The analysis details are broken down into the following:

First Discovered – The time when the sample was first discovered and acquired by Damballa.

Prevalence – The date range when the samples are still being seen in the wild by Damballa.

Infection Vector – How the samples are delivered to the unsuspecting victims.

Symptoms – Observable behaviors in the system that signals the possible presence of malware without actually looking at the registry or searching for the malware file itself.

System Behavior – How the malware works its way through the system to execute its objective.

Network Behavior – A detailed look at how the malware utilizes the domains it connects to.

Protection Mechanism – How the malware hides from the user or system inspection tools.

AV Evasion Techniques – How the malware protects itself from being detected by AV host solutions.

Intent – The main purpose of the malware family

The Command Structure of the Aurora Botnet

	Fake AV Alert / Scareware – Login Software 2009	Fake Microsoft Antispyware Service
Discovered	2009-11-26	2009-08-19
Prevalence	November 2009 – January 2010	August 2009 – September 2009
Infection Vector	Fake AV alerts on compromised or malicious Web sites	Fake AV / Scareware
Symptoms	<ul style="list-style-type: none"> • Login Software 2009 process in startup • Menu Bar and Toolbar of Internet Explorer is missing • System Restore is disabled • Folder Options in Windows Explorer is disabled • Extensions of known file types are hidden • Registry Tools disabled, rendering registry editing inoperable • “Local Settings” folder under “C:\Documents and Settings\<User>\” (where the malware dropper places the dropped and downloaded executables) • Presence of “C:\Documents and Settings\<User>\Windows\system” folder • Pop-up ads • Presence of tracking cookies and displays ads from: <ul style="list-style-type: none"> – counter.surfcouters.com – looksmart.com – maxsun.biz – moreverde.com – oranges88.com – smarttechnik.com – www.prma-enhance.com 	Microsoft Antispyware Services process in startup
System Behavior	Malware propagates through fake malware alerts. The supposed AV installer is actually the malware dropper. Its main purpose is to drop and install the rest of the malware components. Upon execution, it assigns a specific ID to the compromised host, then	Malware propagates through fake malware alerts. The supposed AV installer is actually the malware dropper. Its main purpose is to drop and install the rest of the malware components, typically:

The Command Structure of the Aurora Botnet

registers it to its malware server Web site and downloads the rest of the malware to the compromised host.

To ensure that the malware is downloaded, the creator of this malware dropper uses redundancy in its malware serving Web infrastructure. The dropper checks three different Web sites:

- mcsmc.org
- micronetsys.org
- mnprfix.cn

When Damballa discovered this malware dropper in August 2009, the downloaded executable was version 0. The current version is 3. The functionalities remain similar.

After the successful download of the main component, the main dropper generates a random name and copies the downloaded component to "C:\Documents and Settings\\Local Settings" folder. It calls itself Login Software 2009. The dropped file is then executed to make it active in memory. It survives reboot by autostarting using a common registry entry:

HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

The rest of the components must also be downloaded and executed for them to be active. They are placed in the same folder as the first dropped file. These components create exact copies of themselves with names varying from:

- debug.exe
- mqbxt.exe
- msinits.exe
- win16.exe
- winlogon.exe
- lsass.exe
- drweb.exe
- taskmgr.exe
- win32.exe

- EXE – The component posing as Microsoft Antispyware Services
- VXD – The main dropper downloads and installs ntconf32.vxd, ntsys32.vxd, msmsg32.vxd
- SYS –The main dropper downloads and installs msconfig32.sys

Once the dropper has executed, it can easily bypass UAC since it is given explicit permission by the user, who thought the installation was a real AV product. The first thing the dropper does is to connect to its malware server domain to download its components.

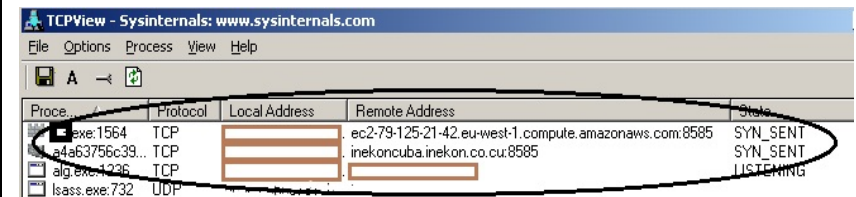
The VXD components are often connected to malware families that have keylogging and spyware behavior. They are also found in some IRC bots. The SYS Component is related to the publicly known and notoriously popular Aurora variant tied to the Google attack.

The EXE component disguises itself as Microsoft Antispyware Services. It runs on Startup using two basic registry keys:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

and

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run



This screen capture shows the dropper attempting to connect to Amazon EC2.

These components are hidden from the user by hiding the folder where they are dropped and changing the attributes of the dropped files to hidden. To survive reboot, these components also are set to autostart using the same technique as the main dropped file.

A DLL file is also dropped in "C:\Windows\System32" with a random filename. Aside from registering (regsvr32.exe) the dropped DLL file to make it active, the malware dropper also modifies the registry to see it as a Browser Helper Object (BHO). It also sets up the DLL to autostart every boot up by using SharedTaskScheduler:

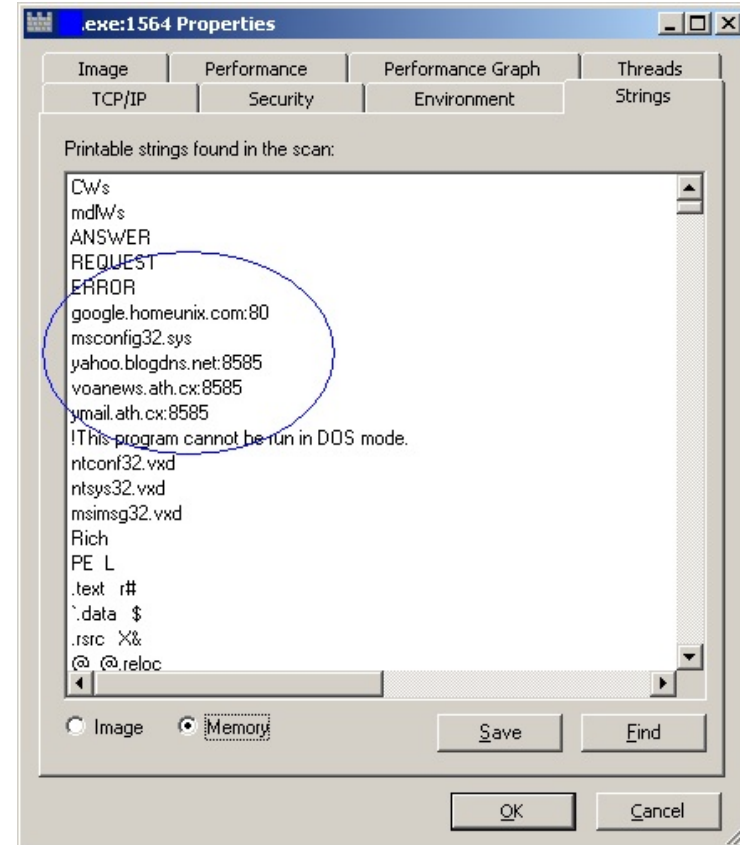
```
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\SharedTaskScheduler
```

This process paves the way for tracking cookies to be downloaded for ads to be served to the compromised host. This DLL is not hidden unlike the other components.

After setting up all the dropped files, the main dropper protects the dropped files by manipulating the settings of Windows Explorer and Internet Explorer. See Protection Mechanism section for more details.

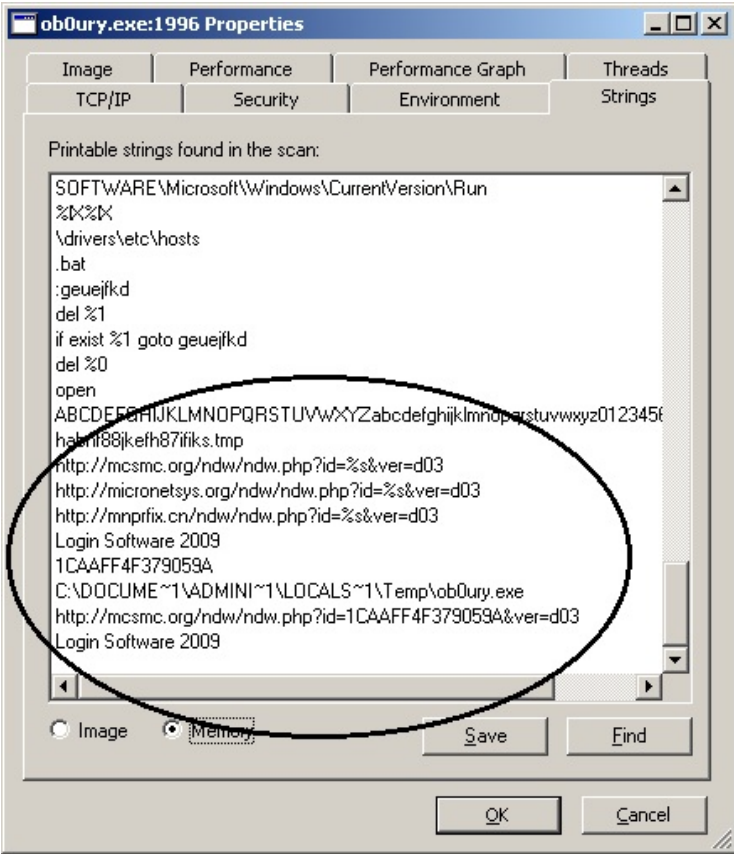
Once all of these "malware installation" tasks are completed by the main dropper, the main dropper activates a batch file to unload itself from memory and deletes both the dropper and the batch file.

The installed malware set is now all active and ready to communicate with CnC.



This screen capture shows a memory string dump that reveals the CnC sought by the EXE malware component.

The Command Structure of the Aurora Botnet

	 <p>This screen capture shows a memory string dump that reveals the CnC sought by the EXE malware component</p>	
<p>Network Behavior</p>	<p>The malware uses domains for two purposes: a malware server domain that hosts the dropped executables and a CnC connection to listen for additional commands.</p>	<p>The malware uses domains for two purposes: malware server domain that hosts the dropped executables and a CnC connection to listen for additional commands. This malware uses Amazon’s EC2 services to serve its malware components.</p>
<p>Malware Server Domains</p>	<p>mcsmc.org micronetsys.org mnprfix.cn</p>	<p>ec2-79-125-21-42.eu-west-1.compute.amazonaws.com ip-173-201-21-161.ip.secureserver.net inekoncuba.inekon.co.cu</p>

The Command Structure of the Aurora Botnet

CnC Domains	<p>filoups.info miecros.info</p> <p>The dropped samples do not listen to the same CnC most of the time. Each listens to a different CnC using a different port.</p>	<p>google.homeunix.com yahoo.blogdns.net voanews.ath.cx ymail.ath.cx</p>
Protection Mechanism	<p>The main dropper also utilizes “Malware Self Preservation” by doing the following before it self-destructs:</p> <ul style="list-style-type: none"> • Hides the location of the malware dropped files by setting the location folder as hidden and the dropped files themselves as hidden. • Disables “Folder Options” in Windows Explorer • Disables “Show hidden files and folders” in Windows Explorer • Hides Internet Explorer’s Menu Bar and Toolbar • Disables System Restore • Disables Registry editing 	<p>None observed.</p>
AV Evasion Techniques	<p>No two dropped files are the same. The malware uses GetTickCount to generate random keys to randomize the hex structure of dropped files.</p> <p>One dropped file (273a51aada271e5a4a91321a3126c767) is packed using FSG v1.3.3.</p>	<p>None observed.</p>
Intent	<p>Money generation through pop-up ads and Web site redirection</p>	<p>Keylogging and spyware.</p>
MD5 Information	<p>Samples Collected/Discovered by Damballa ITW:</p> <p>02677a0770268a20f7ef0d9bd7e8eef1 9803c22252a028b050f6257e7a67d4b7 69ef60094052321d91c0094efd832b92 6e245522d710ca1564e6873a3a0e82bd 0c091b4f6b23b450ccc3d37ccff6cdd6 994a379ff057724248d8435c9be45c1f b5b7146b07b0a0804b36b8056f316722 65510cda14bcefd2419eb1262a6d6829</p>	<p>Samples Collected/Discovered by Damballa ITW:</p> <p>a4a63756c39e345e31f1e8e698ea03a6 2794cacb3f177f340dee0aa2a71bdf1c 2f6c8d68392839cb4615c455cd25fc9c 20ddc972f71c8e584ed2c43254eb811b 1326879b25dd0d7452d7a4b674165a5a</p> <p>(*) denotes that no Rich signature present in the file (^) encrypted</p>

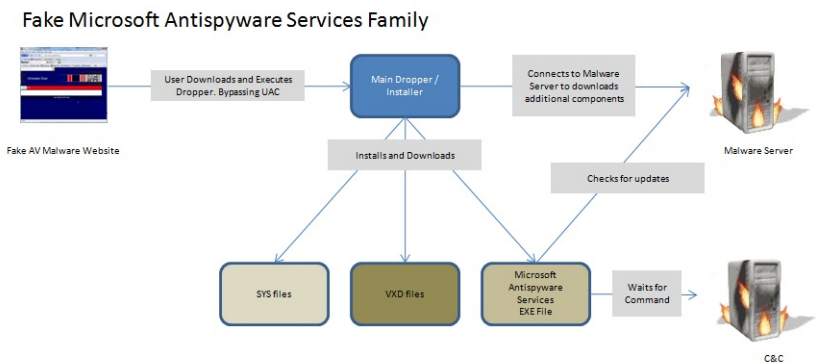
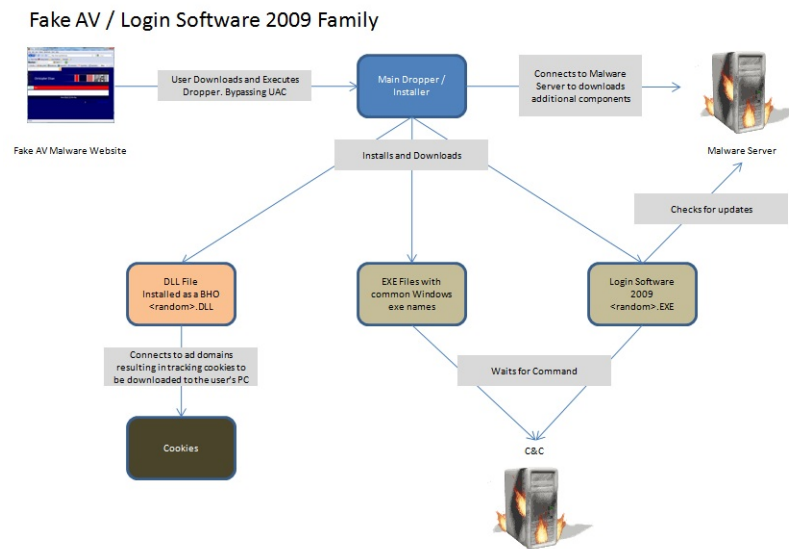
The Command Structure of the Aurora Botnet

01b9c2c916e6d9a82bfc5912348a231f
 0b4872a4f20760739b0007c6b2dc08bd
 253f59417c6c784d6c0e5565736d1815
 273a51aada271e5a4a91321a3126c767*^
 325566e0871ac3d4fccfbb0b4efd8d07
 38ee6476ffe7473707520ef7f5ed5ecb
 62686fd8a1c24abfb7a621e5629ce4ab
 69ef60094052321d91c0094efd832b92
 6e245522d710ca1564e6873a3a0e82bd
 73a88fa854e766d5d3e712db8291bcc8
 863a096685354b2730ad9dfd7e91e942
 b8a177d99854ccc71e94a4a6645e85e7
 d112a2ed6c675158295acb4824b481d8
 feb88ea662de113dcafbe45bdece82fc

(*) denotes that no Rich signature present in the file

(^) encrypted

Malware Diagram



Malware Summary of Findings and Analysis

The predecessor Aurora malware comes from two different families. The newer family came immediately 2 months after the older family, and there was no overlap in their prevalence. For the older family, there was neither an observable protection mechanism nor an AV evasion technique. It was simply a dropper for keylogger files. The newer family has some protection mechanisms and AV evasion techniques. However, it lacks the sophistication found in other botnet malware families.

Below is a summary of the findings of the two malware families that were analyzed.

Common characteristics:

1. Served through fake AV hosting Web sites (no longer available)
2. Common autostart techniques
3. Common older stealth techniques
4. Multiple malware server domains to improve resiliency
5. Droppers and dropped files (EXE and DLLs) were compiled using Microsoft compilers

Differences:

1. Main malware component:
 - a. November 2009 Family – uses DLL file as one of its components
 - b. August 2009 Family – uses VXD and SYS files
2. Main function:
 - a. November 2009 Family – pop-up ads
 - b. August 2009 Family – Suspected keylogger (actual files are no longer available for analysis)
3. Protection Mechanism:
 - a. November 2009 Family – uses basic protection mechanisms to hide itself
 - b. August 2009 Family – none observed

Comparing them to Trojan.Hydraq:

1. Code obfuscation
 - Trojan.Hydraq uses “spaghetti code” in which program elements are separated into small chunks and connected via jump instructions. This technique complicates following the code, and is similar to the tactics employed in old PE viruses that write to small spaces in the host and connect themselves through jump instructions.
 - November 2009 Family – Does not use any code obfuscation. One dropped file is actually packed using FSG v1.33.
 - August 2009 Family – None observed.
2. Autostart Technique
 - Trojan.Hydraq uses Svchost process in Windows by adding its service name in “netsvcs”. When Windows starts, it will load the service into memory.
 - November 2009 Family – Uses common autostart technique using the “Run” key.

- August 2009 Family – Uses common autostart technique using the “Run” key.
3. Intent / Payload
- Trojan.Hydraq – Information gathering
 - November 2009 Family – Pops up ads and Web site redirector
 - August 2009 Family – Information gathering

Malware Significance

Basing on the profile of the two malware families that were analyzed, they are obviously different from each other. The key thing they have in common is that the CnC they utilize are publicly associated with the Aurora botnet.

The botnet controllers preyed on the fear of users that their system is infected with malware. This method saves the botnet controllers from the technical complexity of bypassing Windows’ UAC by using the weakest link in host security – which is the user. The misled user typically clicks OK to everything, bypassing UAC and giving the malware dropper explicit permission to execute.

Neither of the malware predecessor families exhibit the sophistication found in newer malware. Some of the evasion techniques are almost a decade old. Both families use two sets of domains: one for serving malware and the other for CnC.

The droppers and dropped files were compiled using Microsoft Compilers. This is evidenced by the presence of the string “Rich” before the PE header. This watermark is undocumented, meaning there is no mention of this watermark from Microsoft references but they are present in binaries compiled using Microsoft Compilers. Knowing the compiler of choice might help investigators narrow down the individuals or group of individuals responsible for the code.

The simplicity and relative obsolescence of the early versions of the Aurora malware suggest that these malware families were created or written by an individual or group of individuals new to the production of commercial grade malware. Based solely on these families of malwares, it also appears that different individuals or group of individuals created the code:

- The only association the different families have with each other is that they used CnCs associated with Operation Aurora, and they were distributed via similar means. That said, it is possible that two different groups purchased the services of the same crimeware group (probably the same people behind Operation Aurora) to distribute and manage their malware family. Or the crimeware group rented out different variants of the same malware to different groups with different intentions. Price may also be a factor. The less resilient the malware family is, the cheaper it is to purchase or rent.
- The intent of each malware family is different.
- There is no natural progression seen between the two families. Usually malware writers evolve in both technology and protection of their creation but these two families did not show any related evolution. The malware families appear to exist independently, and then become superseded by Trojan.Hydraq.

Piecing it Together

Damballa analyzed network DNS information from a number of distinct and complementary sources ranging from global monitoring systems, enterprise monitoring sensors, passive DNS resolution data

and other DNS streams for this report. At the same time, Damballa also analyzed the malware commonly associated with the Aurora attacks disclosed by Google in January. The result has been a definite correlation between key CnC channels with other malware families that are associated with the criminal operators behind the Aurora botnet.

Based upon our analysis of this attack and the surrounding evidence currently available, we classify the attacks against Google and the other previously identified victim organizations as being typical of current botnet criminal practices. The attack is most notable not for its advanced use of an Internet Explorer 6 Zero-Day exploit, but rather for its unsophisticated design and a pedigree that points to a fast-learning but nevertheless amateur criminal botnet team.

DDNS Findings Summary

Based upon Damballas investigation of DDNS data, the key findings are as follows:

1. The botnet has a simple command topology and makes extensive use of DDNS CnC techniques. The construction of the botnet would be classed as “old-school”, and is rarely used by professional botnet criminal operators any more. However, such reliance upon DDNS CnC is commonly associated with new and amateur botnet operators
2. There were several CnC domains were identified based upon key characteristics of the registration and management of the previously publicly disclosed CnC domains.
3. The major pattern of attacks in mid-December appear to have their origin in July 2009 in mainland China. This likely corresponds to early testing of the botnet CnC.
4. Some of the infections appeared to start within Google’s network. Some of apparent botnet the traffic is not consistent with an IE6/WinXP infection and cannot be easily explained.
5. The attackers had access to large numbers of CnC hosts in geographically diverse hosting co-locations – certainly a high number for a botnet. Further, the botnet used over a dozen domains in diverse DDNS networks for CnC.
6. Only the US victims were compelled to perform MX queries, an event that would typically indicate attempted document exfiltration via email services.
7. Some of the botnets focused on victims outside of Google, suggesting that each domain might have been dedicated to a distinct class or vertical of victims.
8. A review of the TTL period suggests that botmasters de-registered their domains around December 18, 2009.

Passive DNS Data Summary

Based upon analysis of DNS resolution data gathered through a global network of passive DNS monitoring sensors, the key findings are as follows:

1. Cumulative volume of CnC domain name resolutions provides adequate sampling to identify the initialization and growth phases of the Aurora botnet, which also reveals active operation of the CnC channels dating back to June 14th 2009.
2. The victim’s computers connected to, or were part of, 64 different networks, based upon Autonomous Systems (AS) breakdown of Internet netblocks which could represent the upper bound of organizations that may have been breached in the larger Aurora attack. Some organizations (such as Google) own and manage several AS networks. Some of the other AS networks were associated with public Internet Service Providers, which may encompass multiple small and medium businesses.

3. The various CnC domains used by the criminal botnet operators peaked at different times with different rates of lookup by victim systems. These observations correspond to different campaigns run in parallel by different botnet operators and represent the widely publicized attacks that appeared to make use of the Internet Explorer 6 Zero-Day exploit. It is a common tactic by botnet operators to run multiple campaigns at the same time, using different infection vectors (e.g. drive-by downloads, FakeAV, USB infections, etc.) over extended periods of time. This strategy is very consistent with APT campaign methodologies.
4. The vast majority of victim systems appear to have been based in the United States.
5. It is possible to identify the various CnC testing, deployment, management and shutdown phases of the Aurora botnet CnC channels. Some of the CnC domains appear to have been dormant for a period of time after they had infected number victim systems. This type of activity can sometimes be associated with an update to the botnet malware or if the criminal operator sells/trades a segment of the botnet to another criminal operator.

Malware Analysis Summary

Damballa has an array of sources for obtaining new and Zero-Day malware that range from commercial security sharing programs and spam traps to samples gathered from within its enterprise customers' networks. By automatically analyzing tens-of-thousands of new and unique samples each day and extracting their CnC behaviors, Damballa can cluster these malware variants with different botnets. Based upon our analysis of malware samples that relied upon the Aurora CnC domains, our key findings are as follows:

1. The botnet operators behind the Google Aurora attacks deployed other malware families prior to the Trojan.Hydraq release. Some of these releases overlapped with each other.
2. Two additional families of malware (and their evolutionary variants) were identified as "Fake AV Alert / Scareware – Login Software 2009" and "Fake Microsoft Antispyware Service" – both of which were deployed using fake antivirus infection messages to socially engineering the victim into installing the malicious botnet agents.
3. By tracking the evolution of the malware, Damballa was able to identify additional botnet CnC domains used by the criminal operators and establish a timeline of malware associations going back to May 2nd 2009, based upon when a malware sample was captured within an enterprise customer network.
4. Over the time period of this study, the botnet operators improved upon the malware they were deploying. The relative sophistication and armoring of the malware families grow over the months the operators were deploying it, and when they transitioned to entirely new malware families.
5. The major malware families associated with the Aurora botnet attacks are distinct and are unlikely to have been developed by the same malware engineer. This finding is typical of the botnets that Damballa observes targeting enterprise networks. Relatively few botnet criminal operators develop and maintain their own malware. Instead, they typically rely upon third-party contractors or off-the-shelf malware construction kits. As such, core features and functionality changes can occur overnight, but the CnC transitions slowly as the botnet operator ensures that backup CnC domains remain in operation until the victim malware updates (or migration) is complete.

Conclusions

Damballa's findings concerning Operation Aurora can be summarized by the following:

- At the time the attack was first noticed by Google in December 2009, systems within at least 7 countries had already been affected. By the time Google made the public disclosure of the attack on January 12 2010, systems in over 22 countries had been affected and were attempting to contact the CnC servers - the top five countries being the United States, China, Germany, Taiwan and the United Kingdom.
- The Trojan.Hydraq malware, which has been previously identified as the primary malware used by the attackers, is actually a later staging of a series of malware used in the attacks which consisted of at least three different malware 'families'. Two additional families of malware (and their evolutionary variants) have been identified, and they were deployed using fake antivirus infection messages tricking the victim into installing the malicious botnet agents.
- The attacks that eventually targeted Google can be traced back to July 2009, with what appears to be the first testing of the botnet by its criminal operators. The analysis identifies the various CnC testing, deployment, management and shutdown phases of the botnet CnC channels.
- The botnets used dozens of domains in diverse Dynamic DNS networks for CnC. Some of the botnets focused on victims outside of Google, suggesting that each set of domains might have been dedicated to a distinct class or vertical of victims.
- Some of the CnC domains appear to have been dormant for a period of time after they had infected a number of victim systems. This can occur after the botnet operator has updated the botnet malware with new (more powerful) variants or when the criminal operator sells/trades a segment of the botnet to another criminal operator.
- There were network artifacts that suggest that the botnet malware operating with the US-based victims' networks made use of email services to extract the stolen data from the breached organizations.
- There is evidence that there were multiple criminal operators involved, and that the botnet operators were of an amateur level. The botnet has a simple command topology and makes extensive use of Dynamic DNS CnC techniques. The construction of the botnet would be classed as "old-school", and is rarely used by professional botnet criminal operators today

Damballa was able to discover these details on Operation Aurora because of a different approach to researching and neutralizing botnets and other remote-controlled crimeware threats. Command-and-Control – not malware or access point for the attack vector – is the essential element for a successful botnet attack. Everything else about a botnet may change, but CnC must remain in place for the botnet to act in any sort of cohesive manner.

Damballa is the only company that monitors detailed criminal CnC activity within enterprise networks and uses this focus to detect and sever malicious CnC communications. As a result, Damballa has been collecting CnC data for over 4 years, utilizing a globe-spanning array of network sensors within large enterprise customers and Internet Service Provider (ISP) customers. It is this deep visibility into Operation Aurora Cnc that revealed the details in this report.

Although the methods used in Operation Aurora are amateurish and commonplace, the results were just as damaging as a sophisticated botnet because the threat was not quickly identified and neutralized. Aurora's success proves that any breach by a botnet agent, regardless of the quality of the attack vector, is a dangerous security exposure. The result is always hidden and criminal remote

control of enterprise assets, with all of the legal, financial and reputational liabilities that accompany such a serious security lapse.

Additional Reading

"How can I tell if I was infected by aurora", McAfee, 2010,

http://www.mcafee.com/us/local_content/reports/how_can_u_tell.pdf

"Extracting CnC from Malware: The Role of Malware Sample Analysis in Botnet Detection", Damballa, 2009,

http://www.damballa.com/downloads/r_pubs/WP_Malware_Samples_Botnet_Detection.pdf

"Serial Variant Evasion Tactics: Techniques Used to Automatically Bypass Antivirus Technologies", Damballa, 2009,

http://www.damballa.com/downloads/r_pubs/WP_SerialVariantEvasionTactics.pdf

"Botnet Communication Topologies: Understanding the intricacies of botnet Command-and-Control",

Damballa, 2009, http://www.damballa.com/downloads/r_pubs/WP_Botnet_Communications_Primer.pdf

"The Botnet vs. Malware Relationship: The One-to-One Botnet Myth", Damballa, 2009,

http://www.damballa.com/downloads/d_pubs/WP_Botnet_vs_Malware.pdf

"MTrends: The Advanced Persistent Threat", Mandiant, 2010

"Google china cyberattack part of vast espionage campaign, experts say", Washington Post, 2010,

<http://www.washingtonpost.com/wp-dyn/content/article/2010/01/13/AR2010011300359.html>

"Trojan.hydraq", Symantec, 2010, http://www.symantec.com/security_response/writeup.jsp?docid=2010-011114-1830-99

Contributors

Manos Antonakakis

Christopher Elisan

David Dagon

Gunter Ollmann

Erik Wu

About Damballa, Inc.

Damballa stops crimeware threats that exploit enterprise networks for illegal activity by finding and disrupting the hidden communications channels used to control internal servers and hosts. This concentrated focus on malicious remote control delivers fast, accurate insight into advanced network threats, including termination of criminal activity and remediation guidance. Damballa's technology integrates easily with existing infrastructure for cost-effective protection against dangerous security breaches that evade other solutions. The result is smarter, more flexible network security that stops current and future threats, prevents fiduciary breaches and enhances regulatory compliance. Damballa's customers include major banks, Internet service providers, government agencies, educational organizations, manufacturers and other organizations concerned with taking back the command-and-control of their networks. Privately held, Damballa is headquartered in Atlanta, GA.

Copyright © 2010, Damballa, Inc. All rights reserved worldwide.

This page contains the most current trademarks for Damballa, Inc., which include Damballa and the Damballa logo. The absence of a name or logo on this page does not constitute a waiver of any and all intellectual property rights that Damballa, Inc. has established in any of its products, services, names, or logos. All other marks are the property of their respective owners in their corresponding jurisdictions, and are used here in an editorial context, without intent of infringement.