Report

McAfee Labs
Threats Report

April 2017

The Mirai botnet exploited poorly secured IoT devices to perform the largest ever distributed denial-of-service attack.

About McAfee Labs

McAfee Labs is one of the world's leading sources for threat research, threat intelligence, and cybersecurity thought leadership. With data from millions of sensors across key threats vectors—file, web, message, and network—McAfee Labs delivers real-time threat intelligence, critical analysis, and expert thinking to improve protection and reduce risks.

McAfee is now part of Intel Security.

www.mcafee.com/us/mcafee-labs.aspx

Follow McAfee Labs

Introduction

What a winter we have experienced in the world of cybersecurity!

Last fall, cybersecurity crossed into political territory in a major way. In the United States, there were attacks on entities associated with both major political parties, apparently in an attempt to influence the U.S. presidential election. The issue also made it into a debate in that election, too, with one moderator asking "Our institutions are under cyberattack, and our secrets are being stolen. So my question is, who's behind it? And how do we fight it?"

We also saw the first major attack to leverage poor security of Internet of Things devices last fall. The so-called Dyn attack was a distributed denial of service (DDoS) attack that used IoT devices as bots to cripple a major DNS service provider. At its peak, the Dyn attack generated 1.2Tbps of traffic, effectively shutting down many well-known websites. In this Threats Report, we analyze the Mirai malware, which was at the heart of that attack.

The coupling of the importance of cybersecurity in a functioning democracy with the growing awareness of weaknesses in Internet and critical infrastructure security makes it clear that cybersecurity has indeed become a major geopolitical topic.

Just before the holidays, Intel Security held two webcasts on topics familiar to regular readers of McAfee Labs Threats Reports. On December 14, 2016, a panel of seven Intel Security threats researchers discussed our top threats predictions for 2017 and beyond. You can watch the replay of the predictions webcast, and read the McAfee Labs 2017 Threats Predictions report.

A day later, a panel of Intel Security experts discussed the evolution of security operations centers (SOCs). This webcast coincided with the publication of the McAfee Labs Threats Report: December 2016, in which we detailed survey results from a primary research study on the past, present, and future of SOCs. Watch the replay of the SOC webcast.

Last month, Intel Security released the report Building Trust in a Cloudy Sky, which details results from our annual survey about the state of cloud adoption. While trust in the cloud has continued to increase, we also discovered that almost half of those surveyed have slowed their cloud adoption due to a lack of cybersecurity skills.

And finally, Intel Security was quite active at the RSA Conference 2017 in San Francisco. In addition to several product announcements, we announced with our partners the incorporation of the Cyber Threat Alliance. The CTA was established in 2014 as a threat intelligence sharing consortium of IT security vendors. With this incorporation, the cofounding members—Intel Security, Symantec, Palo Alto Networks, Fortinet, Cisco, and Check Point—are now fully committed to the success of the CTA.

The CTA also announced a platform to automatically share and score threat intelligence among CTA members. This platform organizes and structures threat information through "Adversary Playbooks." It provides a way for CTA members to automatically ingest and propagate actionable threat intelligence to better protect customers. Intel Security now uses CTA data to improve protection and detection effectiveness across our threat defense lifecycle solutions.

In this quarterly Threats Report, we highlight two Key Topics:

- We discuss the background and drivers of threat intelligence sharing; various threat intelligence components, sources, and sharing models; how mature security operations can use shared threat intelligence; and five critical threat intelligence–sharing challenges that need to be overcome.
- We examine Mirai, which was responsible for the widely publicized DDoS attack on Dyn, a major DNS service provider. Mirai is notable because it detects and infects poorly secured IoT devices, transforming them into bots to attack its targets.

These two Key Topics are followed by an enhanced set of quarterly threat statistics. New this quarter are several charts that summarize incident activity during the period. We gather data from public sources, our Foundstone Services incident response team, and from our threat research team to create the picture. Incident information is presented over time, by industry sectors and geographies. Let us know what you think.

Soon after the publication of this report, Intel Security is expected to become an independent entity. Late last summer, Intel announced the partial spin-off of Intel Security, which will be known once again as McAfee. Chris Young, Intel Security's Senior Vice President and General Manager since 2014, will become CEO of the new McAfee. Our corporate product strategy will not change. We believe that this change will position McAfee for enhanced focus, innovation, and growth.

And in other news...

In December, as part of our dynamic Endpoint Protection solution, McAfee Labs released Real Protect. This technology detects zero-day malware in near real time, using cloud-based machine learning to automate the classification of suspect malware on the endpoint, based on behavioral and static analysis. We also recently launched McAfee Cloud Threat Detection, which can be used with our web security products and McAfee Network Security Platform. This technology leverages cloud-based machine learning and other group classification methods to identify unknown malware submitted to our cloud and produce analysis reports and indicators of compromise. These are some of the exciting new protection technologies, using machine learning and big data analytics, that power our next generation of protection and intelligence solutions.

Just before the RSA Conference 2017 last month, we went live with enhancements to the Intel Security Threat Center. We now provide a "Threat Landscape Dashboard," which lists the top threats in several categories, including exploit kits, campaigns, ransomware, and vulnerabilities. The dashboard details which threats are significant, their common behaviors, and steps that can be taken to mitigate them. We believe that increased threat awareness leads to better protection. We hope you like it.

Every quarter, we discover new things from the telemetry that flows into McAfee Global Threat Intelligence. The McAfee GTI cloud dashboard allows us to see and analyze real-world attack patterns that lead to better customer protection. This information provides insight into attack volumes that our customers experience. In Q4, our customers saw the following attack volumes:

- McAfee GTI received on average 49.6 billion queries per day in Q4.
- McAfee GTI protections against malicious URLs increased to 66 million per day in Q4 from 57 million per day in Q3.
- McAfee GTI protections against malicious files decreased to 71 million per day in Q4 from 150 million per day in Q3 due to greater download blocking.
- McAfee GTI protections against potentially unwanted programs showed an increase to 37 million per day in Q4 from 32 million per day in Q3.
- McAfee GTI protections against risky IP addresses showed an increase to 35 million per day in Q4 from 27 million per day in Q3.

We continue to receive valuable feedback from our readers through our Threats Report user surveys. If you would like to share your views about this Threats Report, please click here to complete a quick, five-minute survey.

Have a wonderful spring.

—*Vincent Weafer, Vice President, McAfee Labs*

Contents

McAfee Labs Threats Report
April 2017

This report was researched and written by:

Christiaan Beek
Douglas Frosst
Paula Greve
Yashashree Gund
Francisca Moreno
Eric Peterson
Craig Schmugar
Rick Simon
Dan Sommer
Bing Sun
Ravikant Tiwari
Vincent Weafer

Executive Summary

Threat intelligence sharing: What you don't know can hurt you

Sharing threat intelligence significantly reduces attackers' advantages, making their efforts less profitable and shortening the effective lifecycle of campaigns. Historical barriers to sharing are dropping as the legal framework is updated, data standards for interoperability emerge, sharing is automated, and threat intelligence sharing organizations are established.

Threat intelligence sharing promises to improve our ability to protect assets and detect threats. This Key Topic provides a detailed analysis of the background and drivers of threat intelligence sharing, the various components of threat intelligence and its sources, how mature security operations can use this information, five critical challenges that need to be overcome, and the evolving sharing models that have appeared in the market.

To move threat intelligence sharing to the next level of efficiency and effectiveness, we see three areas for improvement:

- We need to simplify event triage and provide a better environment for security practitioners to investigate high-priority threats.
- We need to do a better job establishing relationships between indicators of compromise so that we can understand their connections to attack campaigns.
- We need a better way to share threat intelligence between our own products and with other vendors.

Mirai, the IoT botnet

IoT devices are being hijacked and used to carry out serious crimes in cyberspace. Attackers, after gaining control of IoT devices, can use them to attack business, consumers, or Internet infrastructure. The Mirai botnet is just the beginning.

On October 21, 2016, the domain name service company Dyn was attacked with a massive and complex distributed denial-of-service attack. At its peak, Dyn was flooded by 1.2Tbps of traffic, the highest volume of DDoS traffic ever recorded. The analysis of the attack confirmed that the DDoS traffic originated from Internet of Things devices infected by the Mirai botnet.

Also in October, the source code for Mirai was publicly released. This release has already led to derivative bots, although most appear to be driven by script kiddies and are relatively limited in their impact. The source code release has also led to offerings of "DDoS-as-a-service" based on Mirai, making it simple for willing attackers to execute DDoS attacks that leverage other poorly secured IoT devices.

In this Key Topic, we examine the Mirai botnet and its associated bots, including its architecture and inner workings; its attack process, including the many attack vectors it can use to flood targets; and its evolution.

Key Topics

Threat intelligence sharing: What you don’t know can hurt you

Mirai, the IoT botnet

Share feedback

Threat intelligence sharing: What you don’t know can hurt you

—Vincent Weafer

In a world of constantly changing technology, a world losing defined perimeters and clear areas of trust, traditional security models are under pressure. Increasingly sophisticated attackers are evading discrete defense systems, and siloed systems let in threats that have been stopped elsewhere because they do not share information.

Effective security models must work in the new reality of determining access privileges to secure content from any device, by any user, at any time, from any location, based on multiple attributes that build a more complete picture of the context of the user and the request. Models must also detect and correct evolving threats, from common malware to ransomware, zero-day exploits, and advanced campaigns that use sophisticated planning and technical tools. Although we are improving various security technologies to work more effectively in this new environment, only by sharing intelligence among devices and organizations can we gain the advantage against our adversaries.

In this Key Topic, we discuss the background and drivers of threat intelligence sharing, the various components of threat intelligence and its sources, the sharing models that have appeared in the market, and five critical challenges that we need to overcome. To move threat intelligence sharing to the next level of efficiency and effectiveness, we need to focus on three areas:

- We need to simplify event triage and provide a better environment for security practitioners to investigate high-priority threats.
- We need to do a better job establishing relationships between indicators of compromise so that we can understand their connections to attack campaigns.
- We need a better way to share threat intelligence between our own products and with other vendors.

Why expand threat intelligence sharing now?

Threat intelligence sharing has been around for a long time, driven by security researchers, vendors, and government agencies. Various vertical industry groups have also created intelligence centers for sharing some information about threats targeting their members. This has been a slow and manual process, constrained by publishing calendars, research agendas, and the desire to maximize publicity. Organizations have generally been reluctant to share even the barest details on attacks or compromised systems, in fear of litigation, reputational damage, or publicizing unpatched vulnerabilities. We detailed survey results revealing attitudes and concerns about threat intelligence sharing in the Key Topic “The rise of cyber threat intelligence sharing” in the March 2016 edition of the McAfee Labs Threats Report.

The growing complexity of the technology environment is a very important driver for sharing threat intelligence. Applications, devices, and clouds have enabled anytime, anywhere, anything access demands, making it difficult to distinguish between legitimate and suspicious traffic. Attackers can also come from anywhere at any time, can be highly informed about their targets, and can rapidly adapt to changing circumstances. As a result, we need to know much more about both attacker and target, and better understand the relationships between bits of data to protect the environment.

Further, the growing number and increasing sophistication of attackers has become more and more overwhelming. An entire industry has grown to support these attackers, including subcontractors that can assist in any step of an attack, from malware suppliers to as-a-service providers, from target-list vendors to money launderers. Some feel that legions of "orcs" have gained the upper hand.

Many organizations are coming to realize that the benefits of sharing intelligence now outweigh the disadvantages. The volume and frequency of attacks is so high that anyone and everyone are potential targets, and perhaps the most effective way to contain new attacks is to quickly share timely, contextually rich, machine-readable threat intelligence.

What is threat intelligence?

There are three forms of threat intelligence: tactical, operational, and strategic information.

- **Tactical intelligence** is the information gathered by security systems, scanners, and sensors. Most of this is automated, but is sometimes not communicated or lacks detail. For example, endpoint antivirus systems can detect and block a malicious file, but do not usually provide contextual data such as source IP addresses to decision support systems. This richer trace data would enable centralized systems to identify structural elements that persist through an attack and correlate this info with other data. From a preventive standpoint, the information gathered by these systems are often indicators of compromise, useful for forensic work and remediation efforts, but not detailed or shared quickly enough to protect the entire organization.
- **Operational intelligence** encompasses the critical components for establishing context. Security analysts spend the majority of their time trying to determine which events and alerts should be investigated, the scope and extent of a suspected attack, and how best to coordinate the incident response actions. Too much of this activity is manual, often taking too long to prevent an infiltration or breach. Big data analytics, machine learning, and other automated decision-making techniques are being applied to this problem to augment human capacity and judgment, with the goal of reducing response times and increasing the effectiveness of threat detection and correction.

- **Strategic intelligence** is processed information that informs security policy and planning activities at the organizational level. This includes elements such as the most likely adversaries and their targets, risk probabilities and impact assessments, and regulatory or legal obligations. Although strategic intelligence is largely a manual and experience-driven activity, up-to-date policies, plans, and risk profiles enable faster and more effective responses to suspicious or malicious activities. The net effect is an overall framework of the current cyber threat environment, enhanced by contextual information about specific attacks and threats. This information provides guidance and direction when specific tactics and indicators are detected.

What Is Threat Intelligence?

What activity are we seeing?	Observable	What threats should I look for on my networks and systems, and why?	Indicator
Where has this threat been seen?	Incident	What does it do?	Procedures
What weaknesses does this threat exploit?	Exploit Target	Why does it do this?	Campaign
Who is responsible for this threat?	Threat Actor	What can I do about it?	Course of Action

Models of threat intelligence sharing

A variety of threat intelligence sharing models are already in use, some dating back more than 20 years. Others are evolving to adapt to the changing threat landscape and some are still emerging—as law enforcement, security vendors, government departments, and targeted organizations explore how to effectively share information and respond to the changing regulatory environment.

Information sharing and analysis centers (ISACs)

These are nonprofit organizations that act as centralized collection points and clearing houses for cyber threat intelligence between federal, state, and local governments and specific industry verticals and critical infrastructure sectors. Many of these began as a result of a U.S. presidential directive in 1998 encouraging the sharing of threat information and vulnerabilities between critical infrastructure owners and operators. Although initially focused on U.S. infrastructure, many ISACs have expanded their coverage to include members from around the world. ISACs have also been formed in industries beyond critical infrastructure. Current ISACs serve automotive, aviation, electricity, retail, financial services, nuclear, and water industries, among others.

Information sharing and analysis organizations (ISAOs)

These are more broadly defined than ISACs, as an additional mechanism to encourage and support threat intelligence sharing. Encouraged by a U.S. law passed in 2015 that limited legal liability for sharing threat intelligence with other companies, these organizations can be private or nonprofit, focused on a specific threat or region, and range from communities of interest to government agencies to fee-for-service companies.

Computer emergency response teams (CERTs) and incident response teams (IRTs)

Hundreds of IRTs span the globe. The oldest and best known of these is the CERT at Carnegie Mellon University, in Pittsburgh, established in 1998 at the direction of the U.S. Defense Advanced Research Projects Agency after the Morris worm incident. These organizations provide a range of functions, including threat and vulnerability research, development of some security tools, and coordination of responses to identified threats and vulnerabilities. Most nations have a government-funded CERT or IRT, and most major technology companies operate one focused on their products.

Threat exchanges: sharing for a fee

Threat exchanges are an emerging phenomenon, ranging from for-profit organizations to social network or crowdsourced operations. Like other forms of threat intelligence sharing, these organizations are only as good as their sources and their timeliness. Some multimember or crowdsourced services allow customers to select whom they will share with or rank the trustworthiness of other members.

Marketing: companies publish as a thought leader

Security vendors, technology companies, and other interested parties sometimes plan the release of threat intelligence for its marketing value, for example, to enhance their status as thought leaders. There is a large and growing audience of security practitioners seeking threat reports and blogs. This interest is expanding to reach other interested parties as cyberattacks are increasingly part of the public consciousness. However, as a threat sharing model, reports are generally not up to date enough for tactical protection. Blogs, however, can be a timely and effective communication method for threat intelligence sharing. Although blogs are not machine readable, strategic intelligence is partially formed through this sharing of ideas among practitioners, making these efforts a valuable part of the intelligence sharing model.

Revenue partnerships: commercial threat intelligence and security vendors
Vendor-operated intelligence exchanges offer the advantage of dedicated teams investigating and validating threats.

Critical challenges

Automated threat intelligence sharing is not new but it is still in its early years. During the past several years, the industry has invested in machine generation and machine consumption of tactical threat data. Most data consists of event logs and indicators of compromise such as file hashes, suspicious URLs, and IP addresses. These indicators are very time sensitive, and lose value almost immediately. At the same time, the volume and quality of this data creates new challenges. It is hard to identify high-quality, actionable indicators among the flood of information, making triage difficult for security analysts.

Although the industry has built tactical intelligence sharing capabilities, especially among each company's own products, the industry still fails at sharing high-level, contextually rich intelligence, such as advanced campaigns, at a meaningful level and with other industry participants. Five critical challenges face security vendors and organizations that want to incorporate this valuable intelligence into their security operations. They are volume, validation, quality, speed, and correlation.

Volume

During the past few years, the deployment of enhanced and verbose security sensors and defenses has resulted in a high volume of data fed into threat intelligence tools. Big data analytics and machine-learning tools consume this data and add their analyses to it. The net effect is an improvement in internal capabilities to detect potential attacks and a marked increase in internal threat detection, but a massive signal-to-noise problem remains to be solved. Although the systems are getting better at detection, we have not yet seen a corresponding improvement in the capability of human analysts to triage, process, and act on the intelligence. Vendors are working on solutions to address this problem, from access monitors on sensitive data to sophisticated sandboxes and traps that can resolve contextual clues about a potential attack or suspicious event. Further automation and process orchestration is essential to augment human capacity.

Validation

Disinformation and fake news are not new. Adversaries may file false threat reports to mislead or overwhelm threat intelligence systems. As a result, it is essential to validate the sources of shared threat intelligence, from both inside and outside the organization. Outside validation is perhaps the more obvious requirement, ensuring that incoming threat intelligence is being sent by legitimate sources and has not been tampered with in transit. This is typically accomplished with encryption, hashes, and other methods of digitally signing content. Internal validation is a different problem, not so much validating the sources as analyzing and evaluating the content to determine if it is a legitimate attack or a noisy distraction to draw attention and resources away from a quieter, stealthier threat.

Quality

Related to source validation is the quality of the information we share. Legitimate sources can send anything from definitive indicators of attack or compromise to their entire event feed, which may be of little or no relevance to the receiver. Although more threat intelligence is generally better, much of it is duplicated, and too much low-quality intelligence is of little value. Many threat exchanges are coming online, but they are only as good as their inputs and sensors. Vendors need to re-architect security sensors to capture and communicate richer trace data to help decision-support systems identify key structural elements of a persistent attack. Filters, tags, and deduplication are critical intake tasks to automate in order to increase the value of threat intelligence and make it actionable. An early, promising effort to improve threat intelligence quality will come online in 2017 through the Cyber Threat Alliance. Threat intelligence coming from CTA members will be automatically scored for its quality, and members will be able to draw out threat intelligence only if they have provided sufficient quality input.

Speed

The speed of transmission, or more accurately, the latency between a threat detection and the reception of critical intelligence, is also an important attribute. Intelligence received too late to prevent an attack is still valuable, but only for the cleanup process. This is one reason why open and standardized communication protocols, designed and optimized for sharing threat intelligence, are essential to successful threat intelligence operations. The propagation of attacks between systems happens within a minute or two of a machine's being compromised, so communications between sensors and systems within the enterprise have to operate in near real time. Meanwhile, advanced persistent threats and sophisticated, targeted campaigns often go after multiple organizations in the same vertical market, so communications from one organization to another, usually involving an intermediary or exchange, have to take place within a few hours of the first indication of an attack.

Correlation

Finally, as threat intelligence is received, correlating the information—while looking for patterns and key data points relevant to the organization—is the most critical step. Although some organizations treat the raw data as a proprietary or competitive advantage, the ability to collect data is not a critical factor. It is the processing that turns data first into intelligence and then into knowledge that can inform and direct the security operations teams. The ability to validate data in near real time, correlate it across multiple operating systems, devices, and networks, use it to triage the event, prioritize the investigation, and scope the response is critical to provide effective detection and corrections actions. The goal is to leverage technologies and machine capabilities to triage event data, distill it into high-quality events, and scope and prioritize the incidents so that security analysts can focus their attention on the highest-risk items.

Together, these issues describe threat intelligence sharing's "last mile" problem: taking this information and converting it to controlled action. To cover this last mile we need to find better ways to share threat intelligence between a vendor's products and with other vendors, improve methods to automatically identify relationships between the intelligence collected, and employ machine assistance to simplify triage.

Indicators sitting in a queue, inbox, or report may have arrived at the organizations, but they are of no value until they are deployed into tools that can act on them. An overwhelming volume of threat data waiting to be processed by security analysts is of no value. Security operations across all industries are throwing more and more technology and people at the problem of distilling signals from the noise, turning signals into prioritized incidents, and scoping and investigating those that pose the highest risk.

More technology and resources will not solve the problem without a security and data protection strategy in place. Too many organizations security teams have little idea which pieces of data are the ones that are the most critical to the mission or business, so they attempt a broad-brush approach to protect everything. As the number of devices and the amount of data being generated and stored increases by orders of magnitude, this overload will become unmanageable.

It's time to share

It is critical to collect, triage, and validate data from many sources in near real time and use it to prioritize and scope events. Sharing threat intelligence significantly reduces attackers' advantages, making their efforts less profitable and shortening the effective lifecycle of campaigns. Historical barriers to sharing are dropping as the legal framework is updated. For example, the Cybersecurity Information Sharing Act offers liability protection to private sector organizations and builds the legal foundation for sharing threat intelligence among private organizations and with the U.S. government.

In addition to the CISA, governments are encouraging and supporting the creation of information sharing organizations to speed the collection and distribution of threat intelligence among communities of interest. To effectively use this data, open architectures for the collection and sharing of data are needed so that it is timely and actionable. Data standards for interoperability are emerging, but it is necessary to build automation on top of these protocols that can validate and consume the information at large scale and high speed. Intelligence marketplaces are also emerging, and need to be further enhanced to support and encourage the exchange of information with measurable trust indicators.

The Cyber Threat Alliance is focused on creating a dynamic, real-time trust market for the sharing and monitoring of threat indicators, based on industry best practices. The model is intended to encourage and incentivize companies to share data and fill in the gaps on threat contexts, so that everyone can better protect their systems and limit the effectiveness and duration of an attack.

Cyber Threat Alliance

On February 13, the CEOs of the six founding members of the Cyber Threat Alliance joined to announce the formal launching of the CTA as a standalone entity, with an independent governance structure. This organization is focused on transforming abstract threat intelligence into real-world protections for as many people as possible, as quickly as possible. The CTA platform is an automated intelligence sharing mechanism designed to increase the efficiency and effectiveness of information sharing and enabling threat intelligence to cross that last mile into customer actions. Much of this information will be in the form of adversary playbooks, which structure threat indicators and contextual data into actionable intelligence that enables defenders to focus on protecting their assets.

Attackers test our defenses to find means to evade detection. Enabling multiple vendors and systems to concurrently detect and protect improves the overall effectiveness of all systems, more so than each vendor or system acting in isolation.

Share this Report

Operationalizing Threat Intelligence

To learn more about integrating threat intelligence in an Intel Security environment, read the Operationalizing Threat Intelligence Solution Brief.

The CTA comprises six founding members, which have worked together in various capacities since 2014 (listed alphabetically):

- Check Point
- Cisco
- Fortinet
- Intel Security
- Palo Alto Networks
- Symantec

Additionally, there are multiple contributing members of the CTA. The CTA members believe that their competitive activities are most effective when focused on improving security products and technologies. Jointly creating and sharing a threat intelligence sharing platform will encourage new development and improvements in security protections, ultimately benefiting everyone.

To learn more about integrating threat intelligence in an Intel Security environment, read the Operationalizing Threat Intelligence Solution Brief.

Mirai, the IoT Botnet

—Yashashree Gund, Ravikant Tiwari, and Christiaan Beek

The Internet of Things is transforming everyday devices in ways we are just beginning to appreciate. Already, Intel estimates more than 15 billion smart Internet-connected devices are in operation, and we expect this number will reach more than 200 billion by the end of 2020. Many of these devices are inadequately secured.

On October 21, 2016, the domain name service company Dyn was attacked with a massive and complex distributed denial-of-service (DDoS) attack. The attack took place in two phases. The first took place from 11:10 to 13:20 UTC and the second from 15:50 until 17:00 UTC. At its peak, Dyn was flooded by 1.2Tbps of traffic, the highest volume of DDoS traffic ever recorded. The analysis of the attack confirmed that the DDoS traffic originated from Internet of Things (IoT) devices infected by the Mirai botnet.

The Mirai botnet's apparent author, Anna-Senpai, is thought to be behind the top three DDoS attacks in 2016: the Krebs on Security website, DNS provider Dyn, and French cloud-computing service OVH.

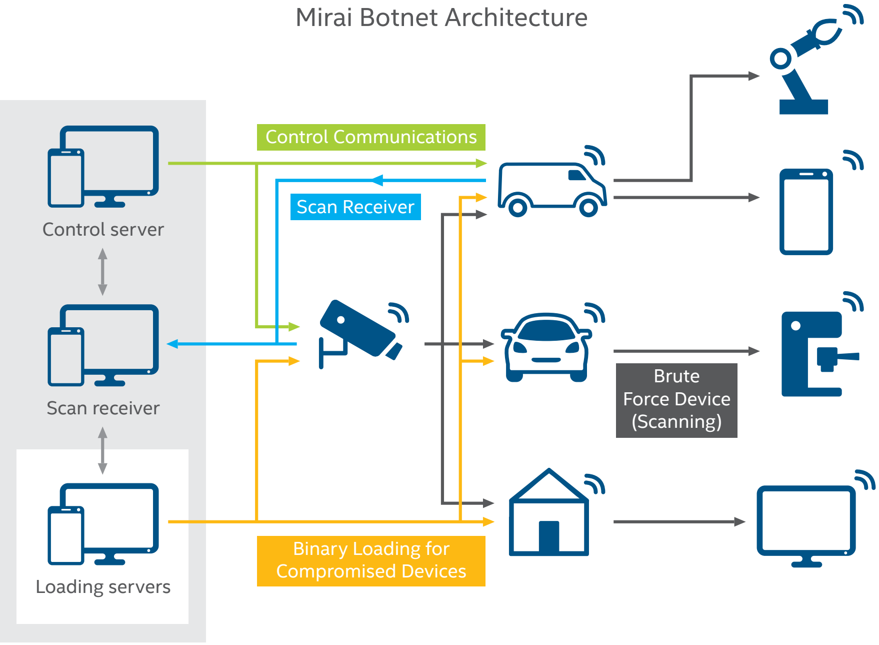
On October 1, 2016, Anna-Senpai publically released the source code for Mirai. (The name comes from the anime series "Mirai Nikki.") This release has already led to derivative bots, although most appear to be driven by script kiddies and are relatively limited in their impact. The release has also led to offerings of "DDoS-as-a-service" based on Mirai, making it simple for willing attackers to execute DDoS attacks leveraging similar poorly secured IoT devices.

In this Key Topic, we examine the Mirai botnet and associated bots, including its architecture and inner workings, the many attack vectors it can use to flood targets, and its evolution.

Architecture and attack process

The Mirai botnet architecture is a relatively straightforward design that includes several remote servers connecting to many instances of the Mirai bot:

Mirai Botnet Architecture



Components

- **Mirai bot**: The main device-resident component, which performs operations such as establishing a connection to the control server, scanning for open Telnet or SSH ports, brute-force testing for default IoT device passwords, and launching DDoS attacks.
- **Control server**: An attacker-controlled server used to monitor successfully infected IoT device bots. The attacker can issue commands—such as launching a DDoS attack—to infected IoT devices using this server.
- **Scan receiver**: Listens for scanned and successfully stolen credentials from IoT devices. It then forwards those credentials to the loading server.
- **Loading server**: Receives stolen credentials from the scan receiver and loads the Mirai bot binary on compromised IoT devices.

Attack process

A Mirai attack starts by scanning a broad range of IP addresses using an advanced SYN scanner for open Telnet or SSH ports and trying to locate IoT devices behind them. SYN scanning determines the state of communications ports without establishing full connections. Mirai then launches a brute-force attack on those IoT devices, using a dictionary of common default usernames and passwords to identify poorly secured IoT devices.

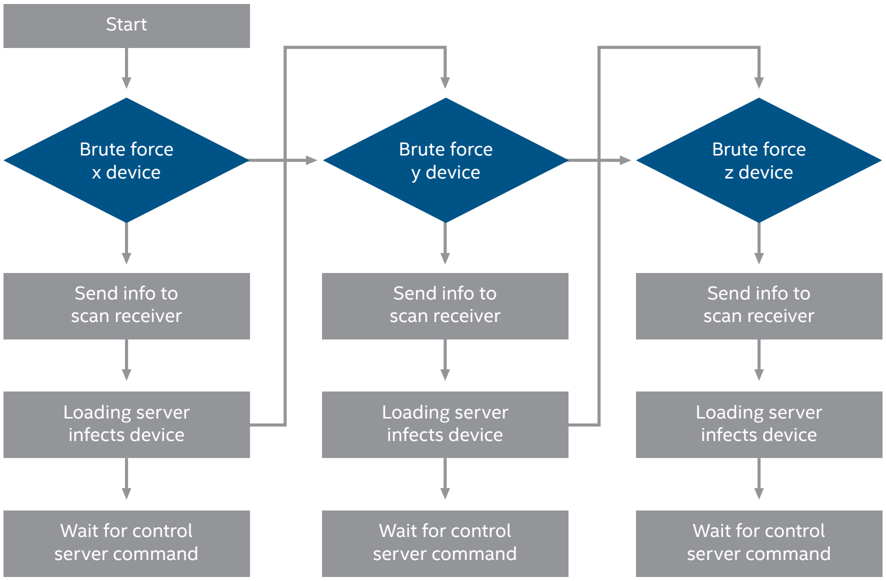
The Mirai bot also resolves the domain name of the control server. Mirai control server domain names are encrypted and hardcoded, and are resolved at runtime to defend against domain name blocking. Once the brute-force attack is successful, the malware sends the compromised IoT device's IP address and credentials to the control server. Alternatively, it may send the information to a dedicated scan receiver server. The IoT device's IP address and credentials are also saved to a database for future use. The scan receiver server forwards the information to multiple loading servers.

Loading servers test each compromised IoT device to determine if the programs wget or tftp are available. If so, the server downloads the Mirai bot binary to the device. If not, it will echo-load a tiny binary (about 1Kb) that acts like wget. The echo-loaded binary downloads the real Mirai bot binary. The loading servers can be configured to use multiple IP addresses to bypass port exhaustion in Linux.

Once compromised IoT devices are successfully infected with the Mirai bot, the preceding process is repeated, with each infected IoT device searching for other vulnerable devices. This method of self-replication was so efficient that Mirai allegedly compromised 380,000 devices solely via Telnet attacks.

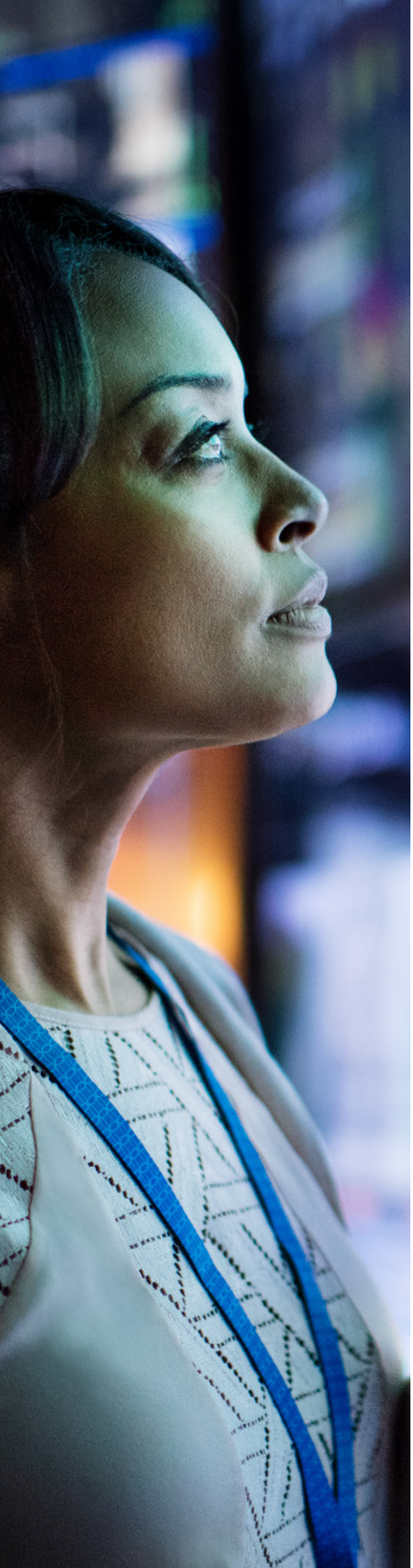
The Mirai bot has implemented additional techniques to guarantee its success. It can kill other bots that may in some way hinder its execution. It can also terminate applications bound to the Telnet or SSH port to give Mirai full and uncontrolled access.

The Mirai Attack Process


Start
Brute force x device
Brute force y device
Brute force z device
Send info to scan receiver
Send info to scan receiver
Send info to scan receiver
Loading server infects device
Loading server infects device
Loading server infects device
Wait for control server command
Wait for control server command
Wait for control server command

Attack vectors

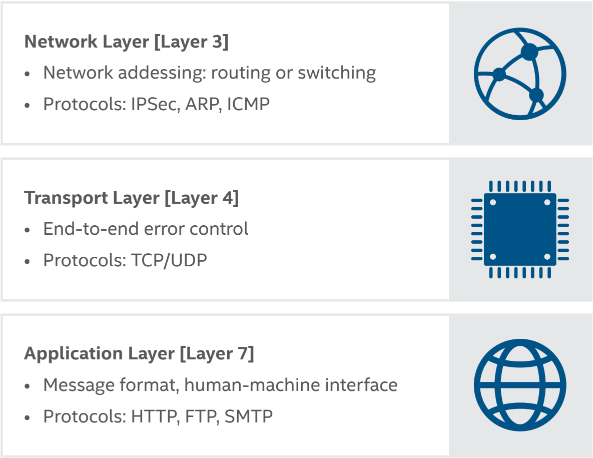
Mirai is capable of DDoS attacks on Layers, 3, 4, and 7 of the OSI model, which standardizes the communications of a computing system without regard to underlying internal structure and technology.



The network layer (3) controls the movement of data from one computer to another. Tasks include logical addressing, routing, datagram encapsulation, fragmentation and reassembly, error handling, and diagnostics.

The transport layer (4) ensures the reliable arrival of messages, providing error-checking mechanisms and data flow controls. The transport layer provides services for both connection-mode and connectionless-mode transmissions.

The application layer (7) is used by network applications. It provides protocols that focus on process-to-process communication across IP networks and provides a firm communication interface and user services.



Network Layer [Layer 3]
- Network addessing: routing or switching
- Protocols: IPSec, ARP, ICMP

Transport Layer [Layer 4]
- End-to-end error control
- Protocols: TCP/UDP

Application Layer [Layer 7]
- Message format, human-machine interface
- Protocols: HTTP, FTP, SMTP

Figure 4: A Mirai attack can exploit three layers of the OSI model.

```
#define ATK_VEC_UDP          0  /* Straight up UDP flood */
#define ATK_VEC_VSE          1  /* Valve Source Engine query flood */
#define ATK_VEC_DNS          2  /* DNS water torture */
#define ATK_VEC_SYN          3  /* SYN flood with options */
#define ATK_VEC_ACK          4  /* ACK flood */
#define ATK_VEC_STOMP        5  /* ACK flood to bypass
                                   mitigation devices */
#define ATK_VEC_GREIP        6  /* GRE IP flood */
#define ATK_VEC_GREETH       7  /* GRE Ethernet flood */
//#define ATK_VEC_PROXY      8  /* Proxy knockback connection */
#define ATK_VEC_UDP_PLAIN    9  /* Plain UDP flood optimized
                                   for speed */
#define ATK_VEC_HTTP         10 /* HTTP layer 7 flood */
```

Figure 5: An attack vector code snippet from Mirai.

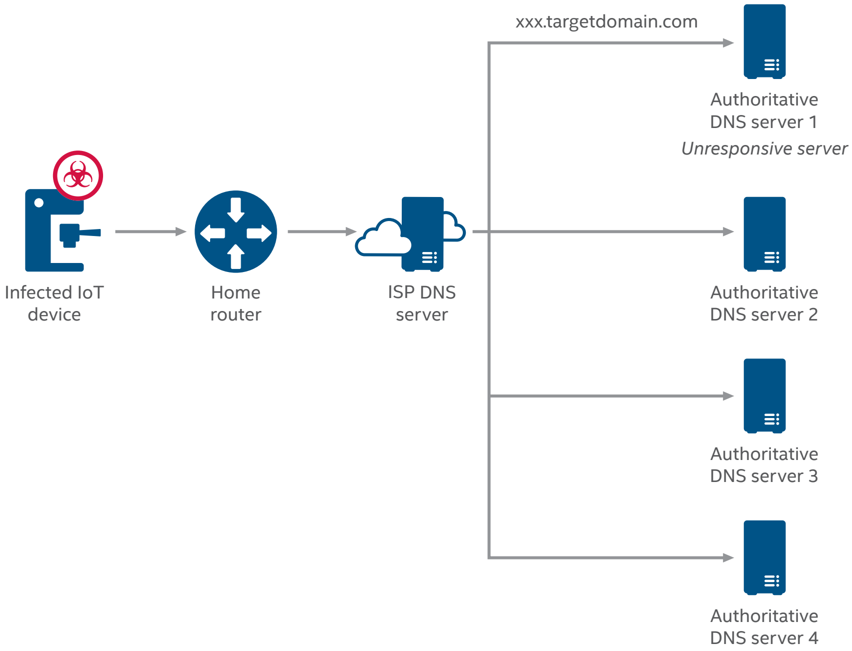
Network-layer attack vectors

- GRE IP and Ethernet flood: Generic routing encapsulation (GRE) is a tunneling protocol that encapsulates packets to route other protocols over IP networks. Sending a large volume of GRE packets to a DNS server consumes resources as the DNS server attempts to de-encapsulate packets. Mirai can perform two type of GRE attacks: with and without Ethernet.

Transport-layer attack vectors

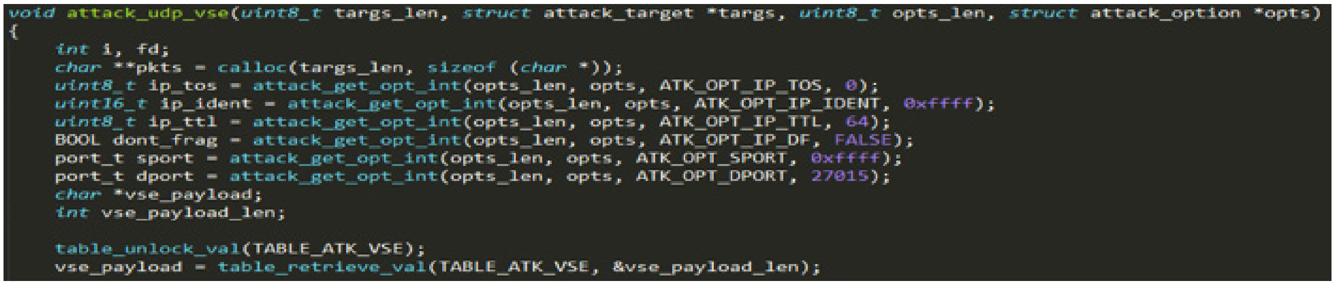
- DNS water torture: This technique tricks an ISP's recursive DNS server into launching an attack on a target's authoritative DNS server. In this technique, infected IoT devices send to the ISP's DNS resolver a small number of well-formed DNS queries that contain the target domain name prefixed with random values (for example, 123.targetdomain.com, 124.targetdomain.com, xxx.targetdomain.com). The ISP's DNS server sends these requests to the target's authoritative DNS server. Once the authoritative DNS server is flooded with such queries from the ISP's DNS server, it becomes unresponsive to the ISP's DNS server. The ISP's DNS server then automatically retransmits the DNS queries to additional authoritative DNS servers.

DNS Water Torture Attack


xxx.targetdomain.com
Authoritative DNS server 1
Unresponsive server
Infected IoT device
Home router
ISP DNS server
Authoritative DNS server 2
Authoritative DNS server 3
Authoritative DNS server 4

- **TCP STOMP flood**: Simple Text Oriented Message Protocol (STOMP) is a simple application-layer, text-based protocol. STOMP is a way for applications to communicate with software developed using different programming languages. The Mirai bot can flood the target with junk STOMP packets. In Mirai's source code we see that each STOMP request is set to 768 bytes, which with the large number of Mirai bots, can result in an attack bandwidth in the gigabytes.

- **TCP SYN flood**: The Mirai botnet sends SYN packets to the DNS server. In response, the DNS server replies with SYN-ACK and waits for an ACK response. The Mirai bot does not reply with an ACK but instead floods the server with multiple SYN requests. For every SYN request, the DNS server waits for the ACK response. This causes the server to become unresponsive as it exhausts all its resources.
- **TCP ACK flood**: The Mirai botnet sends SYN-ACK packets at a very high rate to the DNS server. The DNS server receives a huge volume of SYN-ACK packets out of the normal order of SYN, SYN-ACK, and ACK. The server requires significant processing to understand and respond to this attack traffic. Consequently, the DNS server becomes unresponsive to legitimate traffic.
- **UDP flood**: The Mirai botnet sends a large number of UDP packets to random ports on a DNS server. The victimized DNS server is forced to send many ICMP packets, eventually leading it to be unresponsive to legitimate traffic.
- **UDP VSE queries flood**: This attack works by flooding a gaming server with TSource Engine Query requests. It sends so many requests that the server cannot process all of them and creates a denial of the gaming service. Because Mirai is designed as a service, attackers could use it to promote one gaming industry against its competitors.



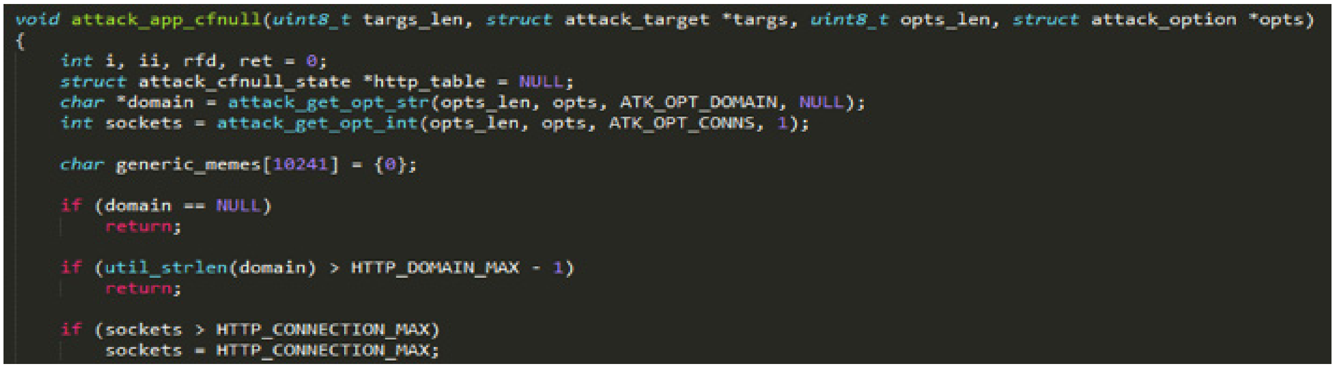
```
void attack_udp_vse(uint8_t targs_len, struct attack_target *targs, uint8_t opts_len, struct attack_option *opts)
{
    int i, fd;
    char **pkts = calloc(targs_len, sizeof (char *));
    uint8_t ip_tos = attack_get_opt_int(opts_len, opts, ATK_OPT_IP_TOS, 0);
    uint16_t ip_ident = attack_get_opt_int(opts_len, opts, ATK_OPT_IP_IDENT, 0xffff);
    uint8_t ip_ttl = attack_get_opt_int(opts_len, opts, ATK_OPT_IP_TTL, 64);
    BOOL dont_frag = attack_get_opt_int(opts_len, opts, ATK_OPT_IP_DF, FALSE);
    port_t sport = attack_get_opt_int(opts_len, opts, ATK_OPT_SPORT, 0xffff);
    port_t dport = attack_get_opt_int(opts_len, opts, ATK_OPT_DPORT, 27015);
    char *vse_payload;
    int vse_payload_len;

    table_unlock_val(TABLE_ATK_VSE);
    vse_payload = table_retrieve_val(TABLE_ATK_VSE, &vse_payload_len);
```

Figure 7: A UDP VSE query flood attack.

Application-layer attack vectors

- **HTTP flood**. The Mirai botnet exploits seemingly legitimate HTTP GET or POST requests to attack a web server.
- **CFNull**: Mirai has an undefined implementation of this attack. A new variant could use this attack simply by defining a vector. Very similar to a GET/POST flood, CFNull sends large payloads of junk, which consume significant server resources.

Share this Report



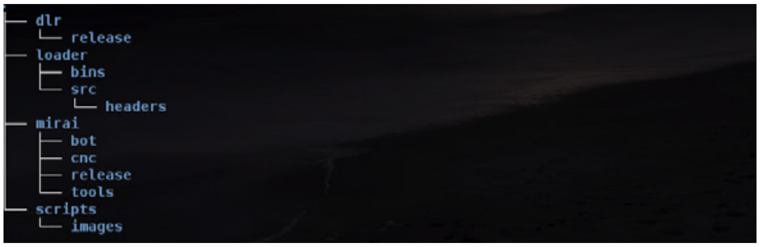
```
void attack_app_cfnull(uint8_t targs_len, struct attack_target *targs, uint8_t opts_len, struct attack_option *opts)
{
    int i, ii, rfd, ret = 0;
    struct attack_cfnull_state *http_table = NULL;
    char *domain = attack_get_opt_str(opts_len, opts, ATK_OPT_DOMAIN, NULL);
    int sockets = attack_get_opt_int(opts_len, opts, ATK_OPT_CONNS, 1);

    char generic_memes[10241] = {0};

    if (domain == NULL)
        return;

    if (util_strlen(domain) > HTTP_DOMAIN_MAX - 1)
        return;

    if (sockets > HTTP_CONNECTION_MAX)
        sockets = HTTP_CONNECTION_MAX;
```

Figure 8: A CFNull attack.

- **Proxy knockback connection**: This attack vector is commented in the Mirai source code but has no current implementation.

Architecture and code analysis

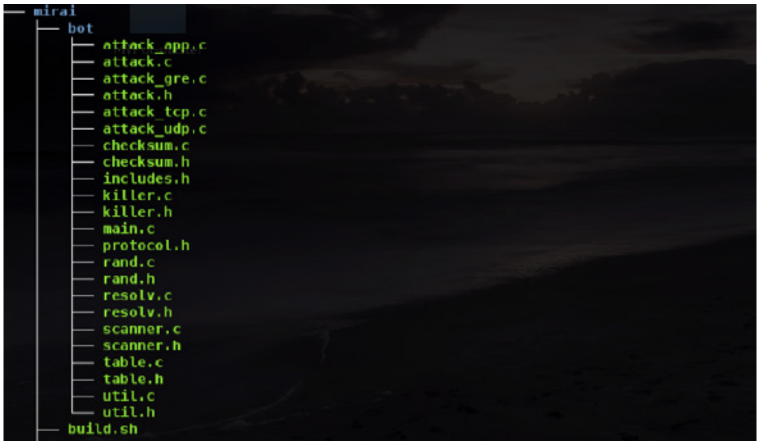
Mirai is Linux-based. It is programmed mainly in C, with the control panel coded in GO (a language developed by Google).



```
├── dlr
│   └── release
├── loader
│   ├── bins
│   └── src
│       └── headers
├── mirai
│   ├── bot
│   ├── cnc
│   ├── release
│   └── tools
└── scripts
    └── images
```

Figure 9: The Mirai botnet code structure.

The Mirai bot is divided into four sections—dlr (downloader), loader, mirai, and scripts. The Mirai bot section contains 13 modules, which perform various tasks.



```
├── mirai
│   ├── bot
│   │   ├── attack_app.c
│   │   ├── attack.c
│   │   ├── attack_gre.c
│   │   ├── attack.h
│   │   ├── attack_tcp.c
│   │   ├── attack_udp.c
│   │   ├── checksum.c
│   │   ├── checksum.h
│   │   ├── includes.h
│   │   ├── killer.c
│   │   ├── killer.h
│   │   ├── main.c
│   │   ├── protocol.h
│   │   ├── rand.c
│   │   ├── rand.h
│   │   ├── resolv.c
│   │   ├── resolv.h
│   │   ├── scanner.c
│   │   ├── scanner.h
│   │   ├── table.c
│   │   ├── table.h
│   │   ├── util.c
│   │   └── util.h
│   ├── build.sh
```

Figure 10: Different modules of the Mirai bot.

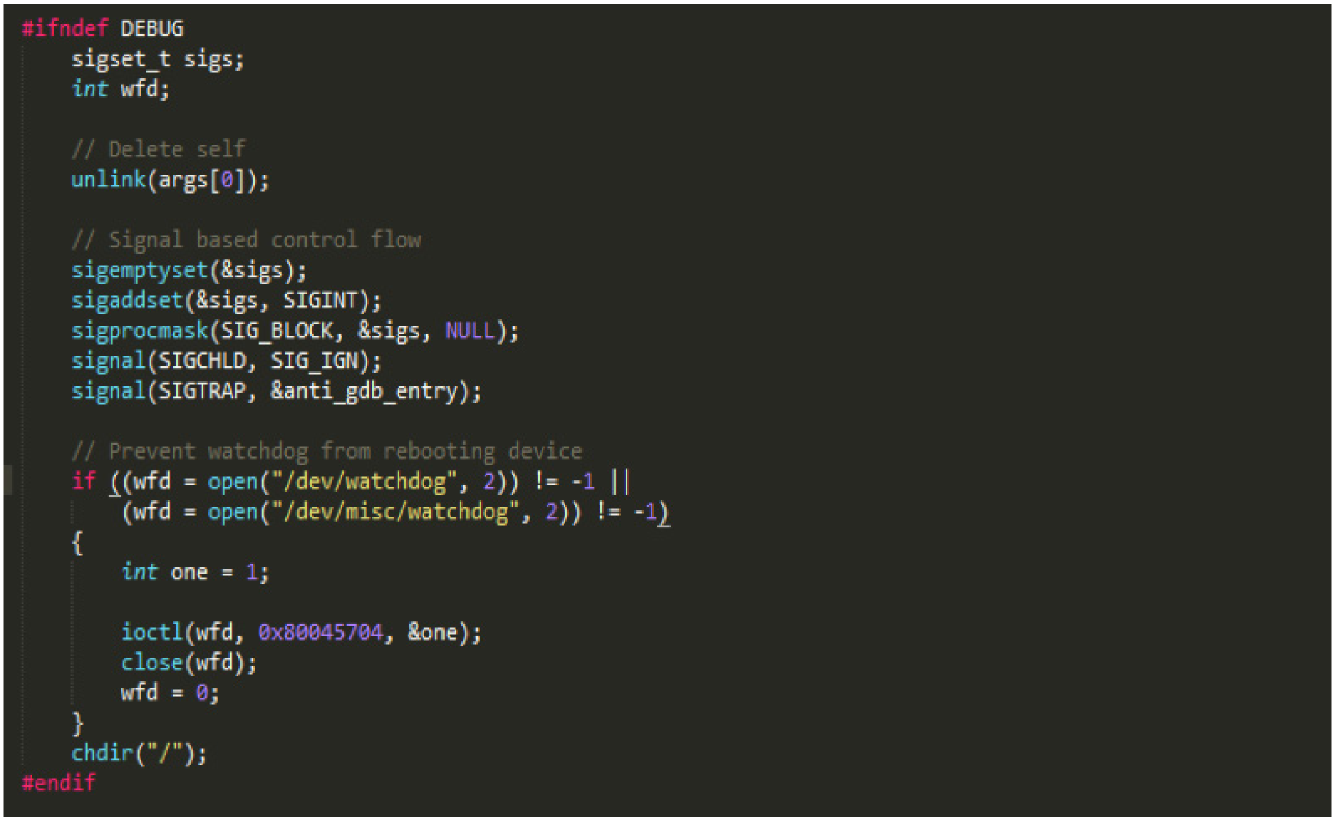
Share this Report

Main module

The Mirai botnet's main module handles almost all the malicious operations, including antidebugging, hiding its own process, control server address parsing, establishing network connections, and DDoS attack execution. This module is also responsible for initializing and calling other modules of the bot.

The module performs the following evasive tasks:

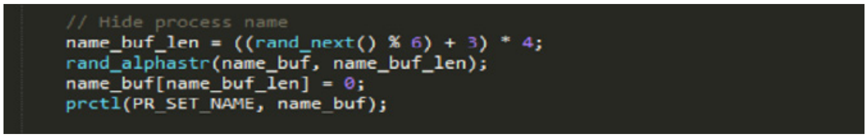
- Unlinks the bot file from the file system (self-deleting) to evade malware forensic operations performed on disk.
- Adds its own handler for breakpoint hits to evade debugging attempts performed on the bot. The breakpoint handler is set to resolve control server addresses.
- Disables the firmware watchdog from rebooting the IoT device in case the device is unresponsive. Once the device is rebooted, the Mirai bot also gets deleted from the IoT device's memory, thus losing control over the IoT device.



```
#ifndef DEBUG
    sigset_t sigs;
    int wfd;

    // Delete self
    unlink(args[0]);

    // Signal based control flow
    sigemptyset(&sigs);
    sigaddset(&sigs, SIGINT);
    sigprocmask(SIG_BLOCK, &sigs, NULL);
    signal(SIGCHLD, SIG_IGN);
    signal(SIGTRAP, &anti_gdb_entry);

    // Prevent watchdog from rebooting device
    if ((wfd = open("/dev/watchdog", 2)) != -1 ||
        (wfd = open("/dev/misc/watchdog", 2)) != -1)
    {
        int one = 1;

        ioctl(wfd, 0x80045704, &one);
        close(wfd);
        wfd = 0;
    }
    chdir("/");
#endif
```

Figure 11: The Mirai botnet's main module.

- Ensures that only a single instance of the bot is running by calling the function ensure_single_instance().
- Hides the process name by replacing the original name with a random name.



```
// Hide process name
name_buf_len = ((rand_next() % 6) + 3) * 4;
rand_alphastr(name_buf, name_buf_len);
name_buf[name_buf_len] = 0;
prctl(PR_SET_NAME, name_buf);
```

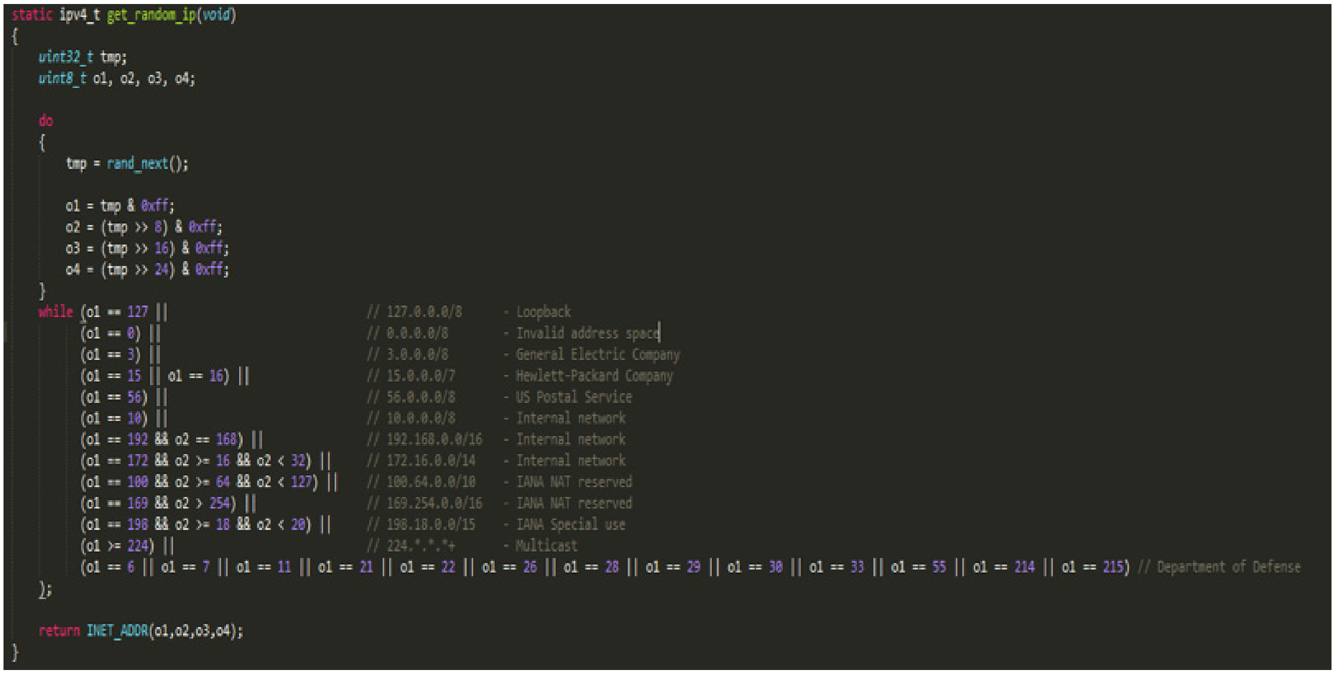
Figure 12: Process name hiding.

- After finishing all initial checks, calls three functions:
  - Scanner
  - Killer
  - Attack

These three functions are called as separate threads without interrupting the main thread. After calling these functions, the main thread starts communicating with the control server.

Scanner module
The scanner module seeks IP addresses to uncover vulnerable IoT devices:

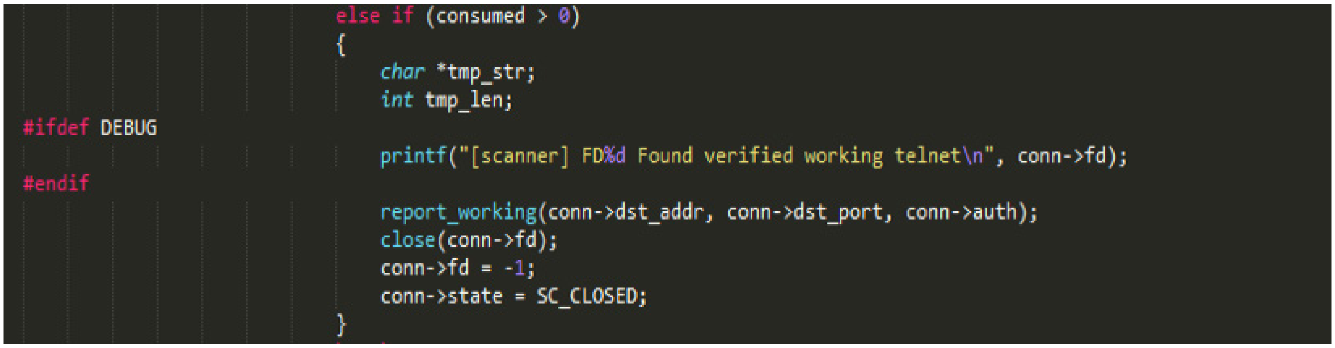
- Creates a separate thread for scanning IoT devices.
- Initializes the parameters required for scanning: credentials, IP headers, etc.
- Uses the SYN and ACK mechanism (SYN scan, brute force) to check for IoT devices.
- Brute-forces Telnet using an advanced SYN scanner.
- Once it receives an IP response, checks a list of IP addresses to avoid.



```
static ipv4_t get_random_ip(void)
{
    uint32_t tmp;
    uint8_t o1, o2, o3, o4;

    do
    {
        tmp = rand_next();

        o1 = tmp & 0xff;
        o2 = (tmp >> 8) & 0xff;
        o3 = (tmp >> 16) & 0xff;
        o4 = (tmp >> 24) & 0xff;
    }
    while (o1 == 127 ||                             // 127.0.0.0/8      - Loopback
          (o1 == 0) ||                              // 0.0.0.0/8        - Invalid address space
          (o1 == 3) ||                              // 3.0.0.0/8        - General Electric Company
          (o1 == 15 || o1 == 16) ||                 // 15.0.0.0/7       - Hewlett-Packard Company
          (o1 == 56) ||                             // 56.0.0.0/8       - US Postal Service
          (o1 == 10) ||                             // 10.0.0.0/8       - Internal network
          (o1 == 192 && o2 == 168) ||               // 192.168.0.0/16   - Internal network
          (o1 == 172 && o2 >= 16 && o2 < 32) ||     // 172.16.0.0/14    - Internal network
          (o1 == 100 && o2 >= 64 && o2 < 127) ||    // 100.64.0.0/10    - IANA NAT reserved
          (o1 == 169 && o2 > 254) ||                // 169.254.0.0/16   - IANA NAT reserved
          (o1 == 198 && o2 >= 18 && o2 < 20) ||     // 198.18.0.0/15    - IANA Special use
          (o1 >= 224) ||                            // 224.*.*.*+       - Multicast
          (o1 == 6 || o1 == 7 || o1 == 11 || o1 == 21 || o1 == 22 || o1 == 26 || o1 == 28 || o1 == 29 || o1 == 30 || o1 == 33 || o1 == 55 || o1 == 214 || o1 == 215) // Department of Defense
    );

    return INET_ADDR(o1,o2,o3,o4);
}
```

Figure 13: IP addresses excluded from scanning.

- After successfully logging in through Telnet, tries to access a shell prompt and reports this information to the scan receiver and loading server for further payload delivery.



```
                        else if (consumed > 0)
                        {
                            char *tmp_str;
                            int tmp_len;
#ifdef DEBUG
                            printf("[scanner] FD%d Found verified working telnet\n", conn->fd);
#endif

                            report_working(conn->dst_addr, conn->dst_port, conn->auth);
                            close(conn->fd);
                            conn->fd = -1;
                            conn->state = SC_CLOSED;
                        }
```

Figure 14: Reporting details of successfully breached devices.

- With an IoT device's IP address in hand, performs a brute-force credential attack. Default username and password combinations include the following:

Share this Report

admin	admin1234
admin	(none)
admin	1111111
admin	1234
admin	12345
admin	54321
admin	123456
admin	admin
root	888888
root	user
root	admin
root	0
root	system
root	pass
root	1111
root	default
root	123456
root	54321
888888	888888

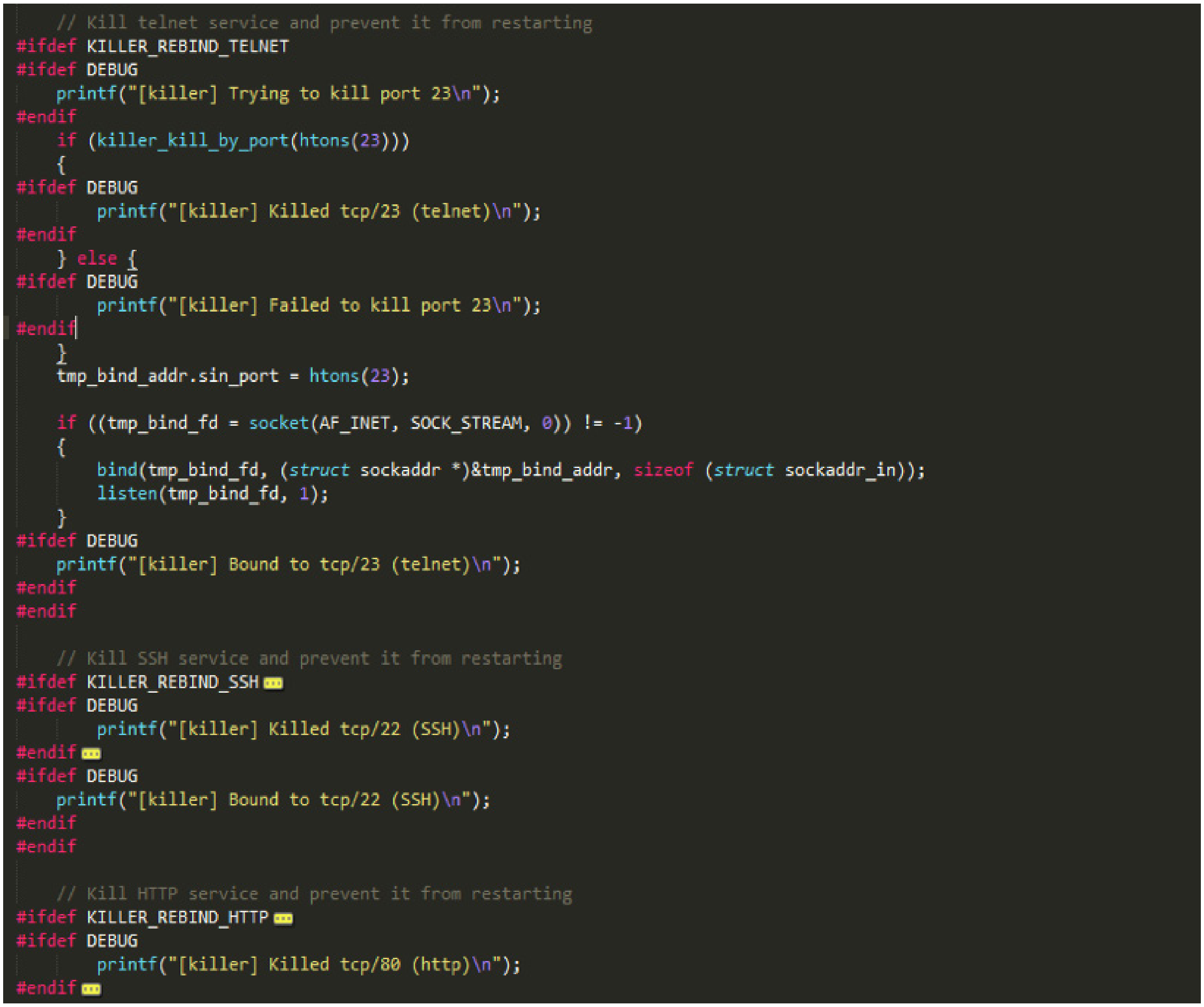
Killer module

The Mirai bot creates a separate thread to stop running processes and services.

- Kills services and processes bound to Telnet, SSH, and HTTP ports, and prevents them from restarting.

Mirai's author anticipated "killing" Telnet on July 10, 2016, on Hackforums. The post "Killing all Telnets" contained the following:

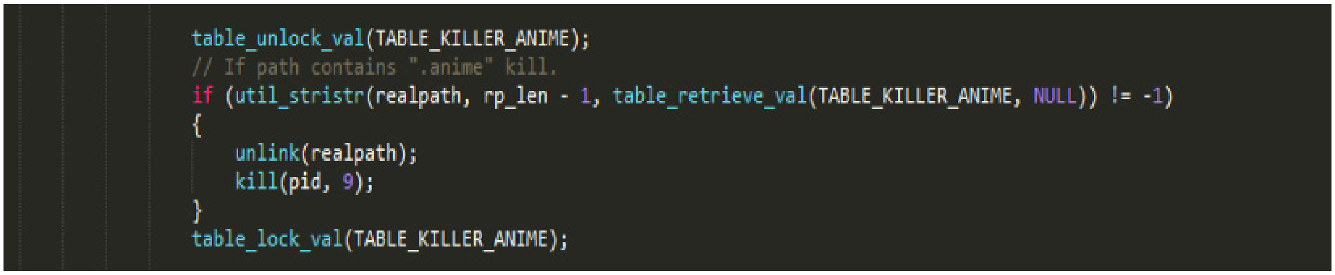
"Just made this post to let you know that as of last night I have begun killing qbots. Watch your botcounts people!"



```
    // Kill telnet service and prevent it from restarting
#ifdef KILLER_REBIND_TELNET
#ifdef DEBUG
    printf("[killer] Trying to kill port 23\n");
#endif
    if (killer_kill_by_port(htons(23)))
    {
#ifdef DEBUG
        printf("[killer] Killed tcp/23 (telnet)\n");
#endif
    } else {
#ifdef DEBUG
        printf("[killer] Failed to kill port 23\n");
#endif
    }
    tmp_bind_addr.sin_port = htons(23);

    if ((tmp_bind_fd = socket(AF_INET, SOCK_STREAM, 0)) != -1)
    {
        bind(tmp_bind_fd, (struct sockaddr *)&tmp_bind_addr, sizeof (struct sockaddr_in));
        listen(tmp_bind_fd, 1);
    }
#ifdef DEBUG
    printf("[killer] Bound to tcp/23 (telnet)\n");
#endif
#endif

    // Kill SSH service and prevent it from restarting
#ifdef KILLER_REBIND_SSH ⋯
#ifdef DEBUG
        printf("[killer] Killed tcp/22 (SSH)\n");
#endif ⋯
#ifdef DEBUG
    printf("[killer] Bound to tcp/22 (SSH)\n");
#endif
#endif

    // Kill HTTP service and prevent it from restarting
#ifdef KILLER_REBIND_HTTP ⋯
#ifdef DEBUG
        printf("[killer] Killed tcp/80 (http)\n");
#endif ⋯
```

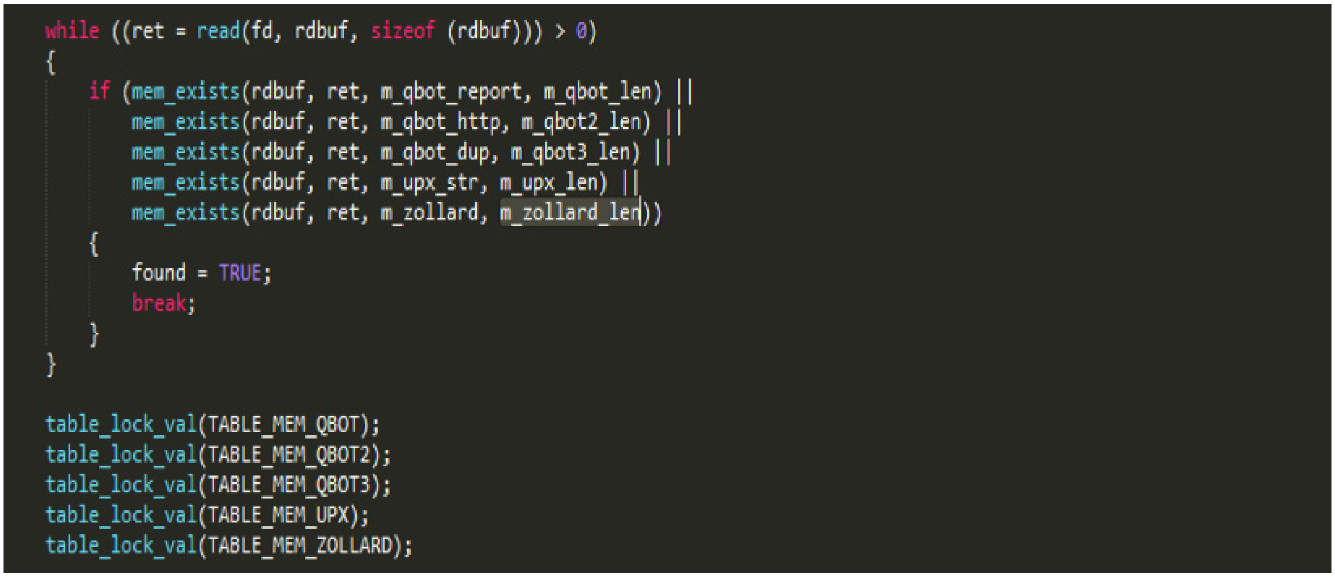
Figure 15: Code snippet to kill processes and services.

- It uses various techniques to detect and remove other malware such as file and process enumeration, as shown in the following screenshots.



```
table_unlock_val(TABLE_KILLER_ANIME);
// If path contains ".anime" kill.
if (util_stristr(realpath, rp_len - 1, table_retrieve_val(TABLE_KILLER_ANIME, NULL)) != -1)
{
    unlink(realpath);
    kill(pid, 9);
}
table_lock_val(TABLE_KILLER_ANIME);
```

Figure 16: File enumeration to kill “Anime” malware.

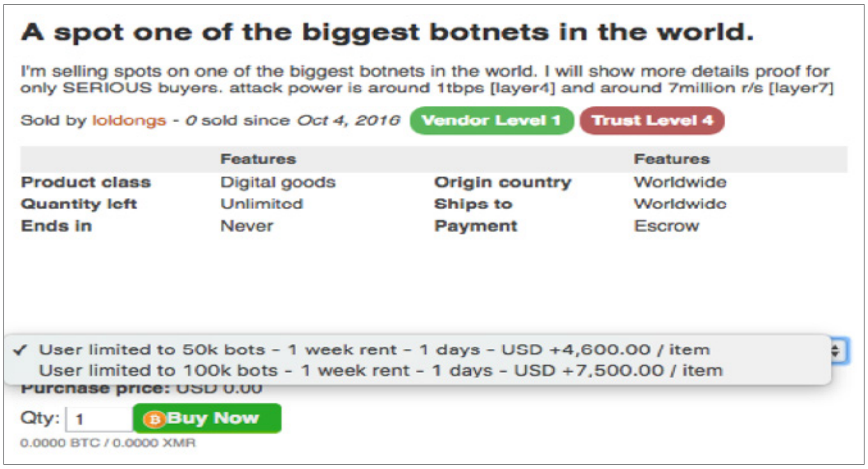


```
while ((ret = read(fd, rdbuf, sizeof (rdbuf))) > 0)
{
    if (mem_exists(rdbuf, ret, m_qbot_report, m_qbot_len) ||
        mem_exists(rdbuf, ret, m_qbot_http, m_qbot2_len) ||
        mem_exists(rdbuf, ret, m_qbot_dup, m_qbot3_len) ||
        mem_exists(rdbuf, ret, m_upx_str, m_upx_len) ||
        mem_exists(rdbuf, ret, m_zollard, m_zollard_len))
    {
        found = TRUE;
        break;
    }
}

table_lock_val(TABLE_MEM_QBOT);
table_lock_val(TABLE_MEM_QBOT2);
table_lock_val(TABLE_MEM_QBOT3);
table_lock_val(TABLE_MEM_UPX);
table_lock_val(TABLE_MEM_ZOLLARD);
```

Figure 17: This process memory scan will detect the presence of other malware such as Zollard or Qbot and kill them.

Botnet-as-a-service

A few days after the release of the Mirai botnet’s source code in October 2016, we found Mirai instances for sale as a service on underground forums and black markets. The first post appeared on October 4 by the seller “loldongs,” who offered Mirai botnet services from US$50 to $7,500. A post on December 25 by “tatsu” offered Mirai rentals for $30 per day.



A spot one of the biggest botnets in the world.

I'm selling spots on one of the biggest botnets in the world. I will show more details proof for only SERIOUS buyers. attack power is around 1tbps [layer4] and around 7million r/s [layer7]

Sold by loldongs - *0* sold since *Oct 4, 2016* Vendor Level 1 Trust Level 4

	Features		Features
Product class	Digital goods	**Origin country**	Worldwide
Quantity left	Unlimited	**Ships to**	Worldwide
Ends in	Never	**Payment**	Escrow

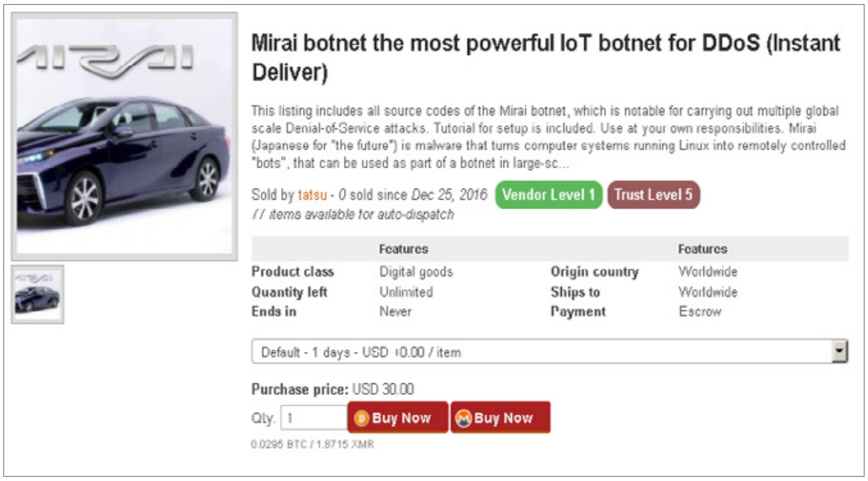
✓ User limited to 50k bots - 1 week rent - 1 days - USD +4,600.00 / item
User limited to 100k bots - 1 week rent - 1 days - USD +7,500.00 / item

Purchase price: USD 0.00

Qty: 1 Buy Now

0.0000 BTC / 0.0000 XMR

Figure 18: Mirai for sale on Alpha Bay.



Mirai botnet the most powerful IoT botnet for DDoS (Instant Deliver)

This listing includes all source codes of the Mirai botnet, which is notable for carrying out multiple global scale Denial-of-Service attacks. Tutorial for setup is included. Use at your own responsibilities. Mirai (Japanese for "the future") is malware that turns computer systems running Linux into remotely controlled "bots", that can be used as part of a botnet in large-sc...

Sold by tatsu - *0* sold since *Dec 25, 2016* Vendor Level 1 Trust Level 5
// items available for auto-dispatch

	Features		Features
Product class	Digital goods	**Origin country**	Worldwide
Quantity left	Unlimited	**Ships to**	Worldwide
Ends in	Never	**Payment**	Escrow

Default - 1 days - USD +0.00 / item

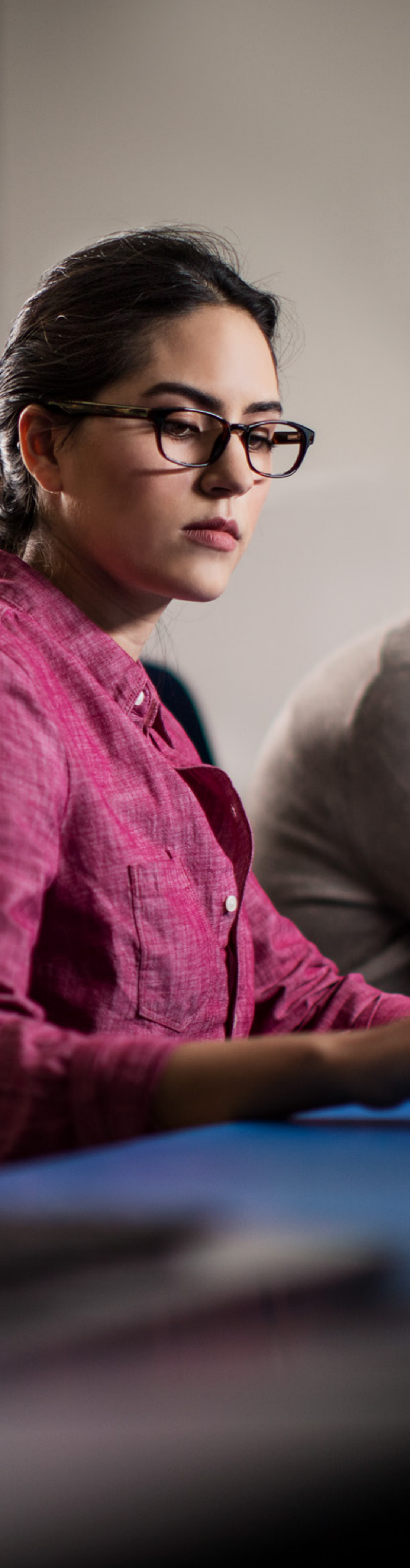
Purchase price: USD 30.00

Qty. 1 Buy Now Buy Now

0.0295 BTC / 1.8715 XMR

Figure 19: Another post on Alpha Bay selling Mirai.

To enslave more poorly secured IoT devices for use by a DDoS-as-a-service offering, attackers are using Mirai in combination with new vulnerabilities. One example of this combination is Mirai with a remote code execution vulnerability in the TR-069 configuration protocol. Mirai has also evolved in other ways to expand and maintain its ecosystem of IoT bots.



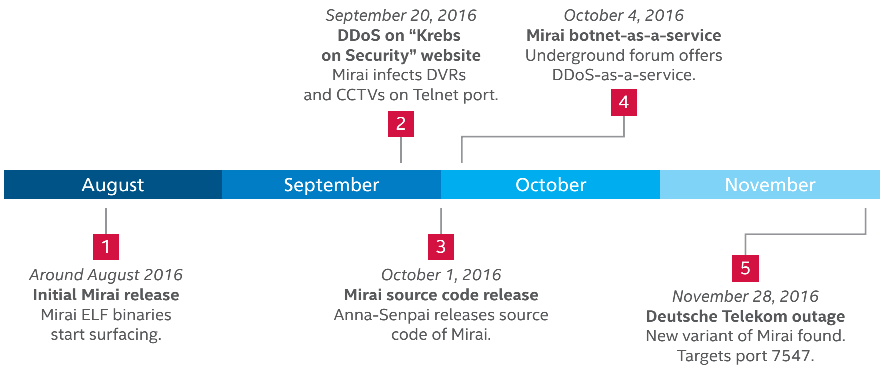
Evolution

Mirai has evolved to enslave additional IoT devices in different categories and from different vendors. In the beginning, Mirai infected online DVRs and closed-circuit cameras.

Gradually, new Mirai variants were found in the wild. On November 28, 2016, Deutsche Telekom reported an outage caused by a new IoT device worm. This worm was confirmed as a new variant of the Mirai bot that uses the TR-064 and TR-069 protocols over port 7547 to exploit a known vulnerability and gain control of the IoT device. Protocol TR-069 runs “provisional networks” used by ISPs and telecoms to remotely manage modems and routers in their consumer networks. The new Mirai variant exploits these provisional networks further to spread the Mirai bot within a modem’s or router’s network “segment,” which can vary in size from a street to a municipality or even an entire country.

Hardcoded control server domain names were replaced by domain generation algorithms. Using this technique, domain names are dynamically generated with a very short lifespan, making any attempt to sinkhole or block these domains very difficult.

Mirai Evolution Timeline



1 *Around August 2016* **Initial Mirai release** Mirai ELF binaries start surfacing.

2 *September 20, 2016* **DDoS on “Krebs on Security” website** Mirai infects DVRs and CCTVs on Telnet port.

3 *October 1, 2016* **Mirai source code release** Anna-Senpai releases source code of Mirai.

4 *October 4, 2016* **Mirai botnet-as-a-service** Underground forum offers DDoS-as-a-service.

5 *November 28, 2016* **Deutsche Telekom outage** New variant of Mirai found. Targets port 7547.

Implications of the source code release

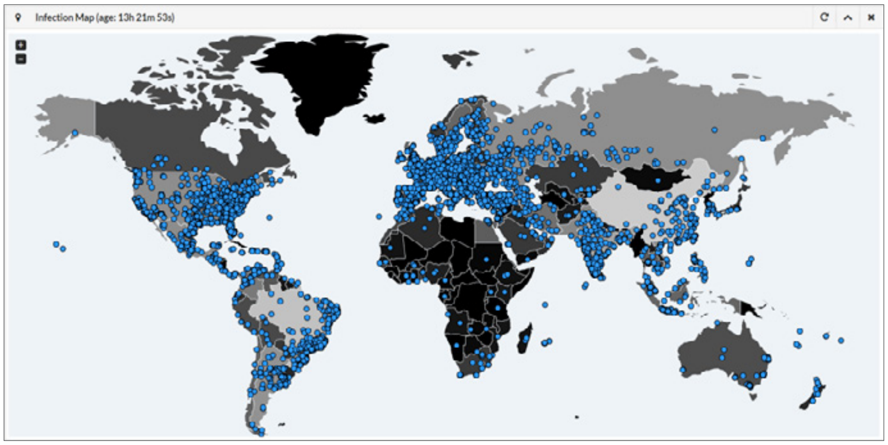
The Mirai source code release will facilitate new Mirai variants with more advanced features. It may also serve as a backbone for completely new IoT device malware. There will likely be a rise in attackers such as script kiddies who will use Mirai to launch DDoS attacks against critical services on the Internet.

We also anticipate that many DDoS attackers will choose Mirai as their malware. Professional attackers will combine Mirai with advanced antidefense countermeasures to thwart the detection of the bot on IoT devices.

Who uses Mirai?

After the source code was released, many attackers began creating their own Mirai bot variants. Security analysts face several questions: How big is the problem, what motivates attackers, what are their targets, and can we profile them?

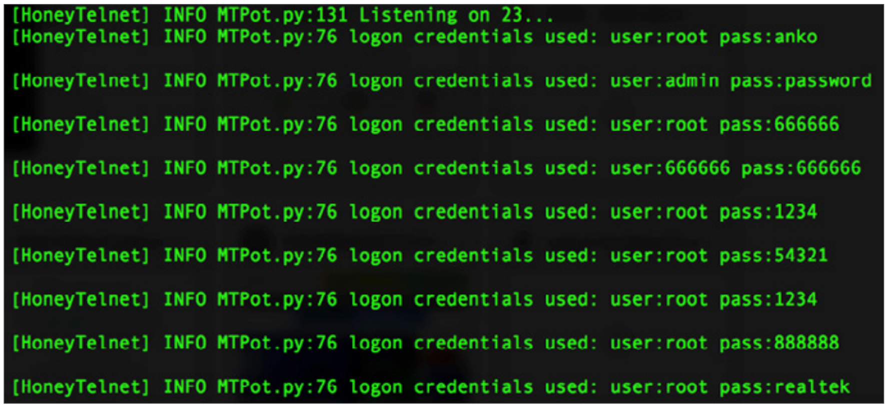
A source for tracking Mirai botnets is on this webpage.



When we took the preceding screenshot, there were about 40,000 infected IoT devices online and about 2.5 million IoT devices offline, which could mean they were shut down or had been cleaned of the malware. Further, we saw that every minute, about five IP addresses were added to Mirai botnets.

As we saw in the source-code analysis, the Mirai bot contains a list of default passwords commonly used by DVRs, IP cameras, and routers. To see how fast one device can be infected, we set up a honeypot simulating a vulnerable IoT device and hosted it at a random data-hosting provider.

In fewer than five minutes, we registered the first attempted attacks:



```
[HoneyTelnet] INFO MTPot.py:131 Listening on 23...
[HoneyTelnet] INFO MTPot.py:76 logon credentials used: user:root pass:anko

[HoneyTelnet] INFO MTPot.py:76 logon credentials used: user:admin pass:password

[HoneyTelnet] INFO MTPot.py:76 logon credentials used: user:root pass:666666

[HoneyTelnet] INFO MTPot.py:76 logon credentials used: user:666666 pass:666666

[HoneyTelnet] INFO MTPot.py:76 logon credentials used: user:root pass:1234

[HoneyTelnet] INFO MTPot.py:76 logon credentials used: user:root pass:54321

[HoneyTelnet] INFO MTPot.py:76 logon credentials used: user:root pass:1234

[HoneyTelnet] INFO MTPot.py:76 logon credentials used: user:root pass:888888

[HoneyTelnet] INFO MTPot.py:76 logon credentials used: user:root pass:realtek
```

Figure 22: Login attempts by the Mirai botnet on our dummy IoT device.

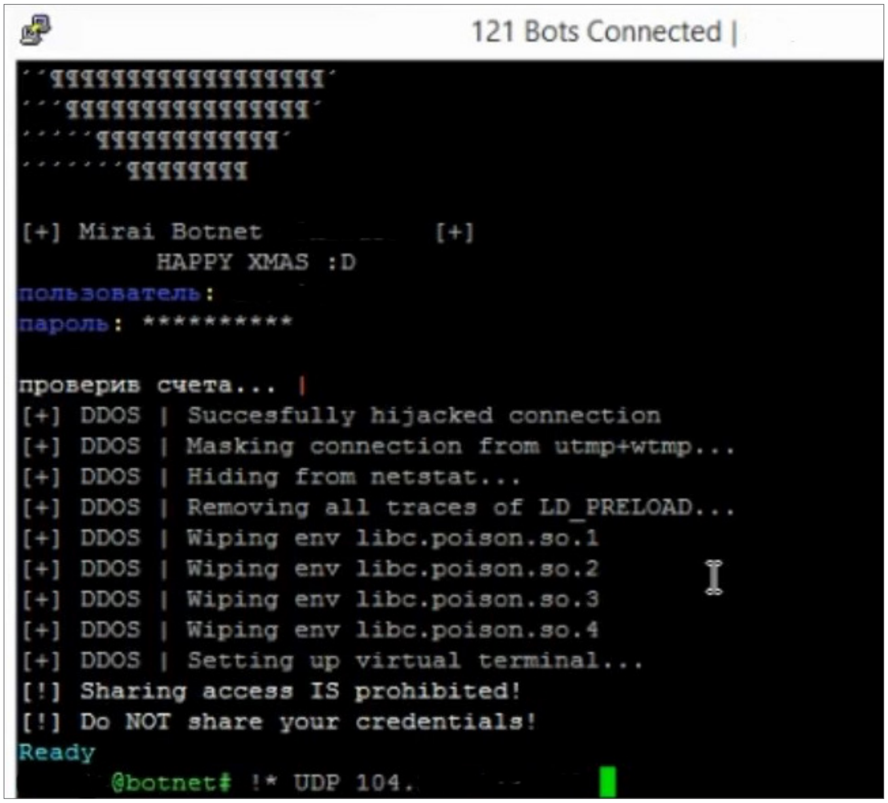
Watch a video of the honeypot console, showing how quickly the simulated IoT device was discovered and attacked.

We monitored some of the feeds and our honeypot to understand the kind of attacks executed and who was targeted. Over two days, we observed a total of 34 DDoS attacks executed by multiple Mirai botnets. The most common attack vector was the "GRE IP flood," followed by the "ACK" and "STOMP" floods. Analyzing the data over a longer period, the most common attack vectors were "SYN flood" and "UDP flood."

The IP addresses under attack were mostly hosted in the United States, with a few in the Netherlands, the United Kingdom, and Germany. Examining the targeted victims, we discovered that they were mostly gaming servers, a few individuals, a web shop, and a dating site. Even a "competitor" DDoS attack site was attacked:

Figure 23: Attacking DDoS and stressor competitor sites is very common in this line of business.

Our analysis suggests that a different audience is now using the Mirai code. Exploring further, we found several tutorials that teach script kiddies how to set up a Mirai botnet so they can attack their friends or other targets. The tutorial videos illustrate that amateur attackers are gathering more botnet code and are playing with it, not realizing its power. In one tutorial, the presenter live-streamed his setup of a Mirai botnet. After receiving help from a couple of his buddies over Skype, he successfully installed the code, tested it with 121 IoT device bots connected and then launched a UDP flood attack against his victim:



```
121 Bots Connected |

´´¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶´
´´´¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶´
´´´´´¶¶¶¶¶¶¶¶¶¶¶¶´
´´´´´´´¶¶¶¶¶¶¶¶

[+] Mirai Botnet [redacted]     [+]
          HAPPY XMAS :D
пользователь: [redacted]
пароль: **********

проверив счета... |
[+] DDOS | Succesfully hijacked connection
[+] DDOS | Masking connection from utmp+wtmp...
[+] DDOS | Hiding from netstat...
[+] DDOS | Removing all traces of LD_PRELOAD...
[+] DDOS | Wiping env libc.poison.so.1
[+] DDOS | Wiping env libc.poison.so.2
[+] DDOS | Wiping env libc.poison.so.3
[+] DDOS | Wiping env libc.poison.so.4
[+] DDOS | Setting up virtual terminal...
[!] Sharing access IS prohibited!
[!] Do NOT share your credentials!
Ready
[redacted]@botnet# !* UDP 104.[redacted]
```

The future of IoT threats

IoT devices will continue to be hijacked and will be used to carry out serious crimes in cyberspace. Attackers, after gaining control of IoT devices, could render them unusable for their owners by changing the firmware's login credentials. IoT malware currently uses default credentials to gain control of devices, the easiest path. Once that door closes, attackers will find other entry points. IoT security problems will not disappear by simply changing default passwords.

IoT malware is now very basic, but this will change with the emergence of more professional and well-funded attackers. As a result, we can expect more advanced IoT bots. Intel's Matthew Rosenquist foresees several attack vectors, including:

- Exploiting vulnerabilities in real-time operating systems, especially targeting industries that are dependent on IoT devices.
- More advanced bypassing mechanisms such as encrypted communication, peer networks, and more complex control structures.

Actionable policies and procedures for securing IoT devices

- Research the IoT device's security track record. Before buying an IoT device, see if it, or the company offering it, has had problems. A quick Internet search might suffice. A search of the Federal Trade Commission's website will reveal prior enforcement actions. By doing some research, you may find some companies ignore their product's security concerns, while others are more proactive.

Share this Report

Technical Brief

Secure IoT Devices to Protect Against Attacks

To learn how Intel Security products can help protect against botnets, click here.

- Keep all IoT device software up to date. This simple best practice can often remove vulnerabilities, especially those recently discovered and publicly highlighted. Have a patching procedure in place and verify if the patches have been applied successfully.
- For IoT devices that cannot be patched, mitigate the risk by leveraging application whitelisting, which locks down systems and prevents unapproved program execution.
- Segment IoT devices from other parts of the network using a firewall or intrusion prevention system. Disable unnecessary services or ports on these systems to reduce exposure to possible entry points of infection. Mirai exploits unused ports.
- Change defaults and use strong passwords. Default and weak passwords are the main threat to IoT devices. Adopt good password habits, such as using long phrases, special characters, mixed cases, and digits. Passwords must be strong and not easy to guess.
- Take advantage of IoT security settings. Some IoT devices will offer advanced configurations, and you should make the most of them. Certain IoT products may offer separated networking, similar to a guest Wi-Fi network alongside your main connection. That is just one feature—more may come with other products.
- Connect IoT devices using secure Wi-Fi. Create strong passwords and use the latest security protocols, such as WPA2.
- Restrict physical access to IoT devices. Direct device tampering can also lead to IoT device hacks.
- Disable Universal Plug and Play (UPnP) support. Many IoT devices support UPnP, which makes the device discoverable on the Internet and vulnerable to malware infections. Disable this when feasible.
- Power-cycle IoT devices periodically. Malware is commonly stored in a volatile memory and can be erased by shutting off and restarting the device.

To learn how Intel Security products can help protect against botnets, click here.

Threats Statistics

Malware

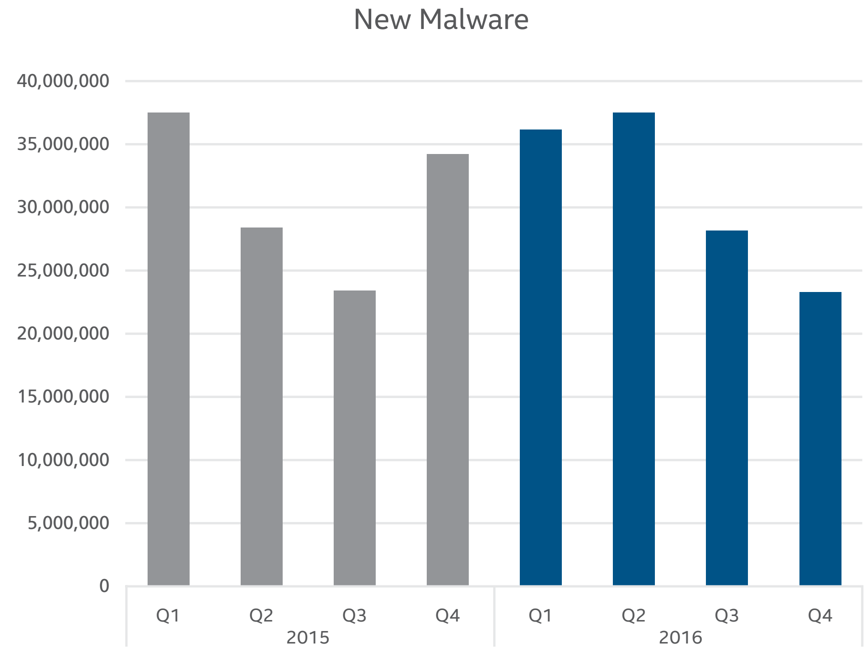
Incidents

Web Threats

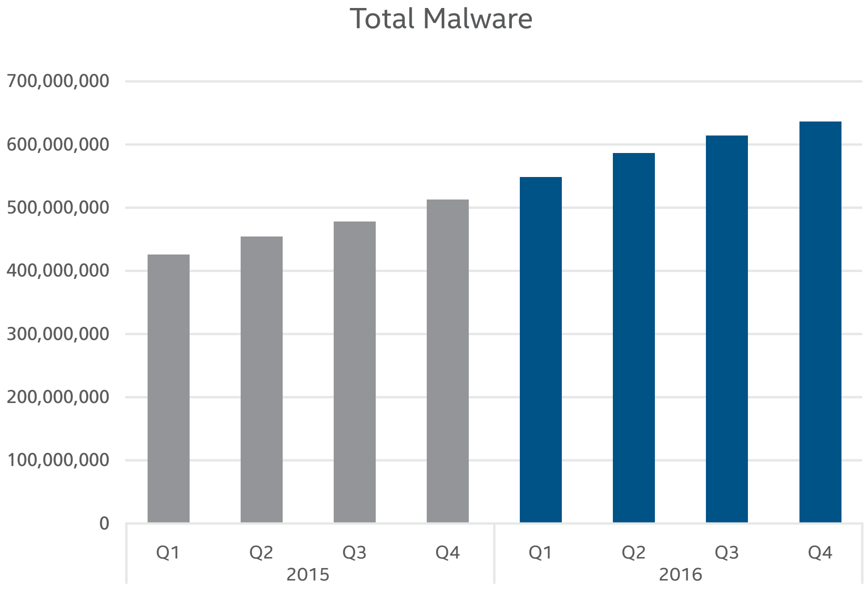
Share feedback

Malware

The volume of malicious samples cataloged per quarter ebbs and flows quarterly and annually. The decline during the past three quarters mirrors the trend we observed at the start of 2015. A pattern of two to three quarters of growth followed by three quarters of decline has been consistent since 2013.

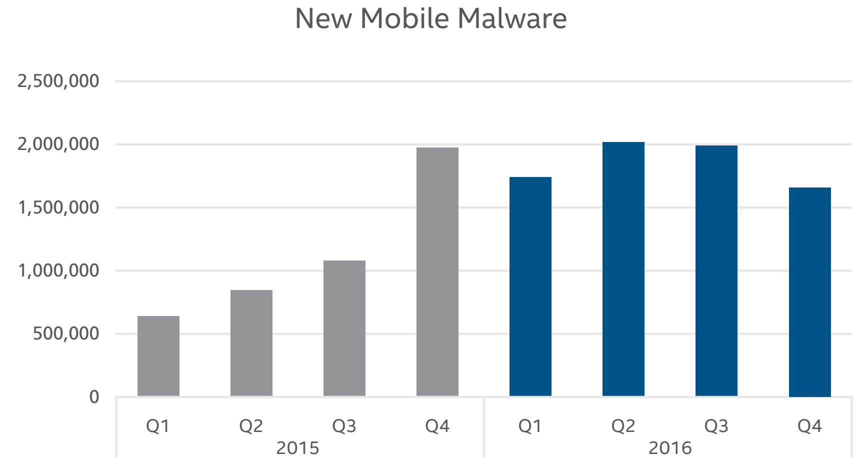


Source: McAfee Labs, 2017.

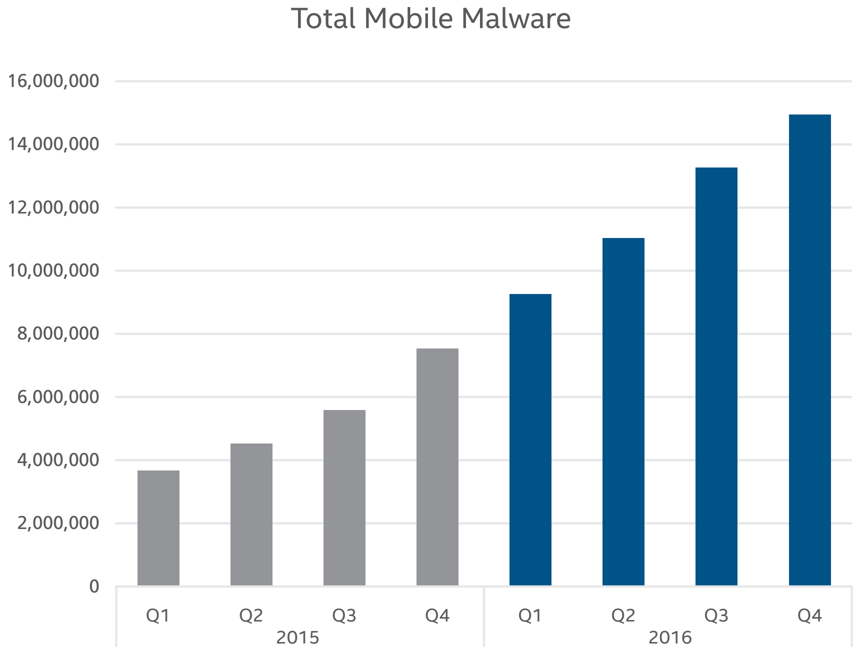


Source: McAfee Labs, 2017.

Share this Report



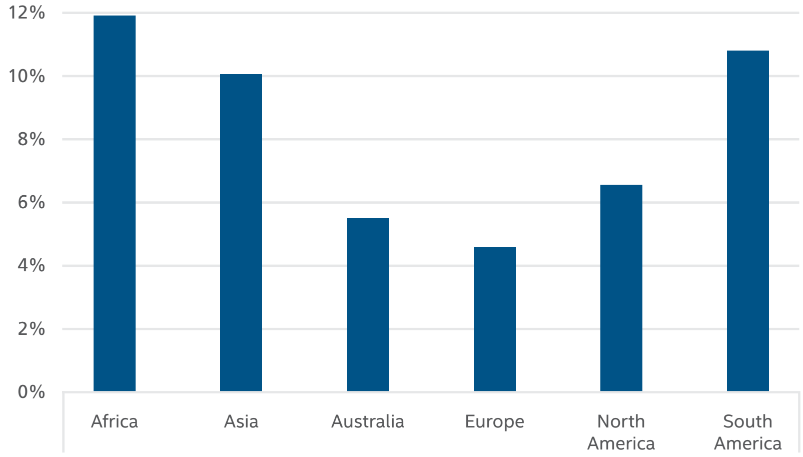
Source: McAfee Labs, 2017.



Source: McAfee Labs, 2017.

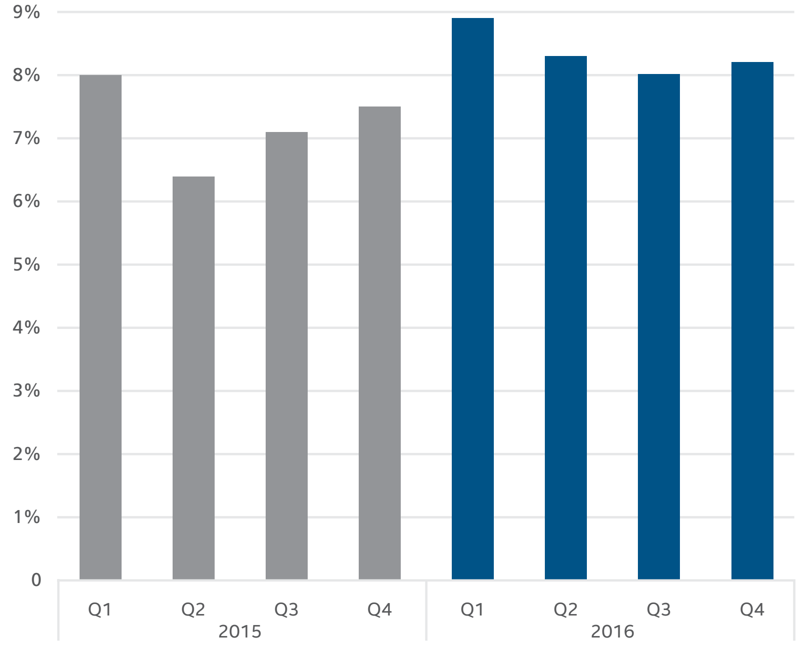
Share this Report

Regional Mobile Malware Infection Rates in Q4 2016
(percentage of mobile customers reporting infections)



Source: McAfee Labs, 2017.

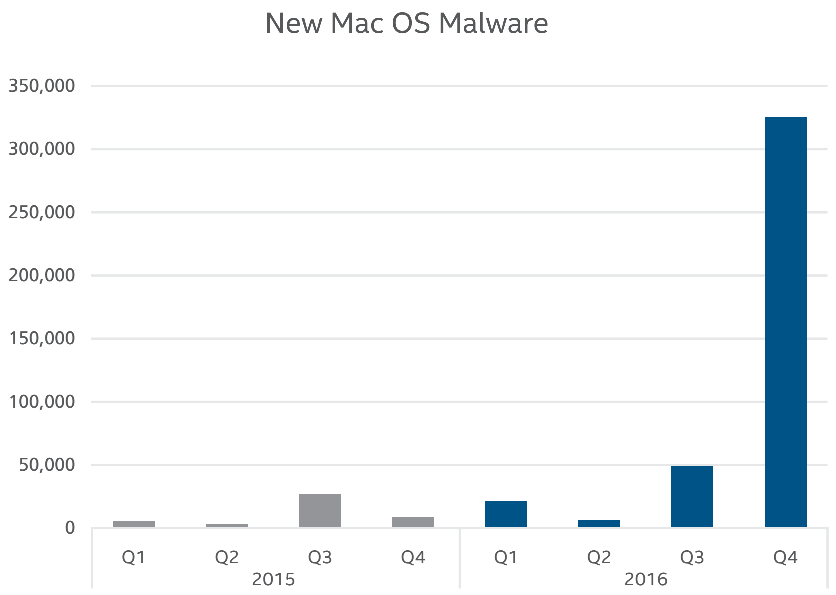
Global Mobile Malware Infection Rates
(percentage of mobile customers reporting infections)



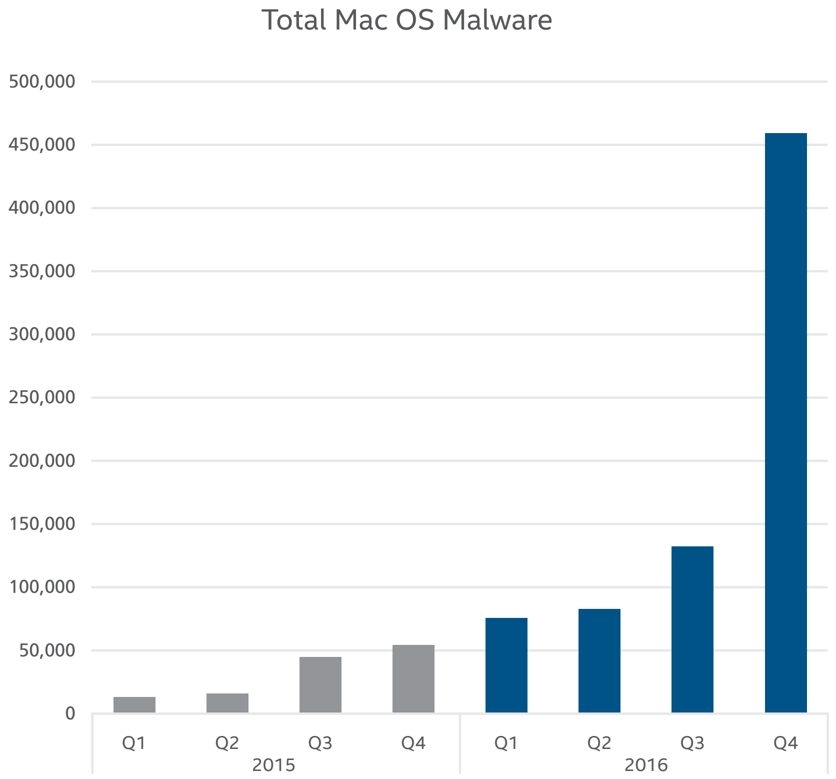
Source: McAfee Labs, 2017.

Share this Report

Just as last quarter, the big increase in Mac OS malware was due to adware bundling.



Source: McAfee Labs, 2017.

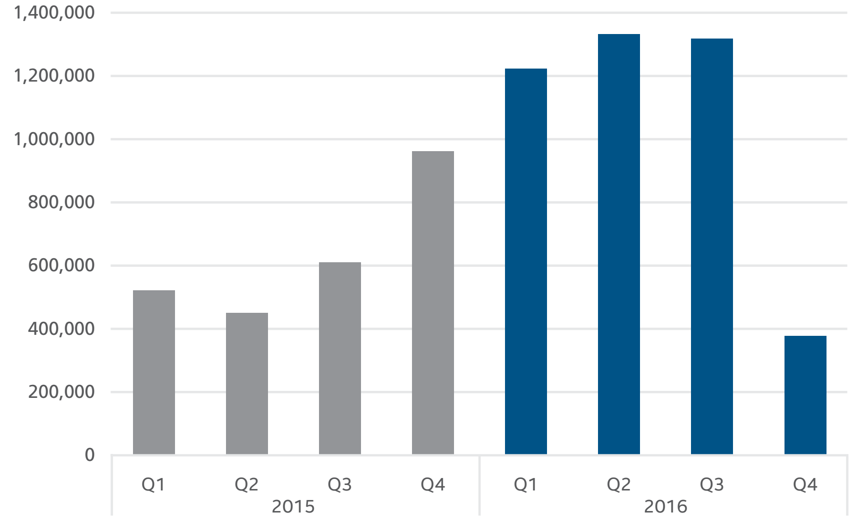


Source: McAfee Labs, 2017.

Share this Report

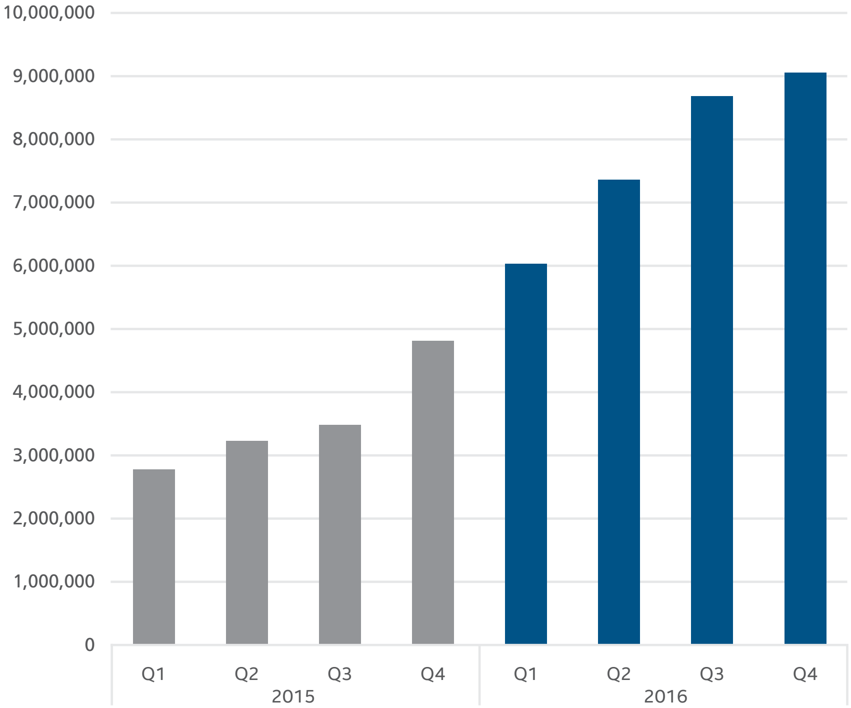
The big decline in ransomware is mostly due to a drop in generic ransomware detections, as well as a decrease in Locky and CryptoWall.

New Ransomware



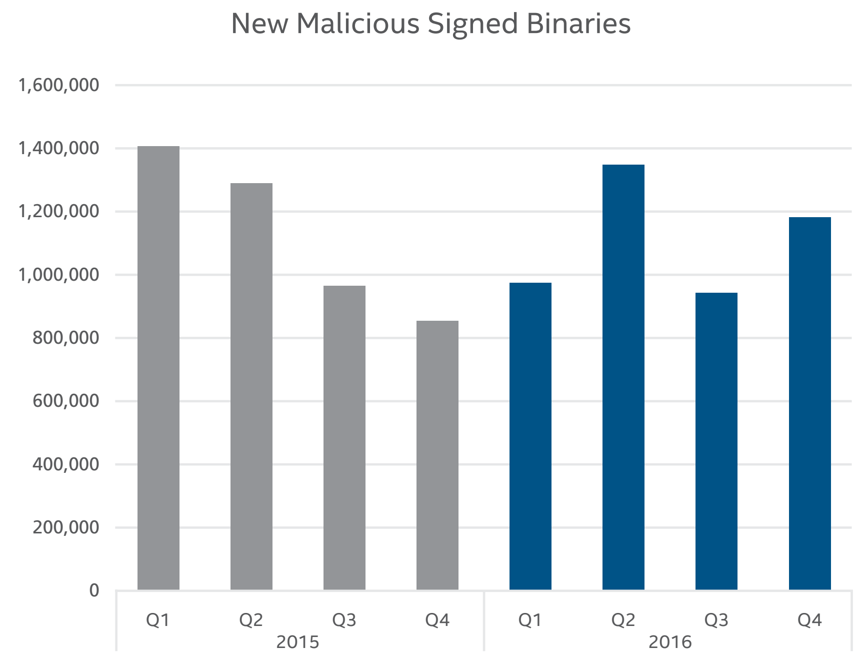
Source: McAfee Labs, 2017.

Total New Ransomware

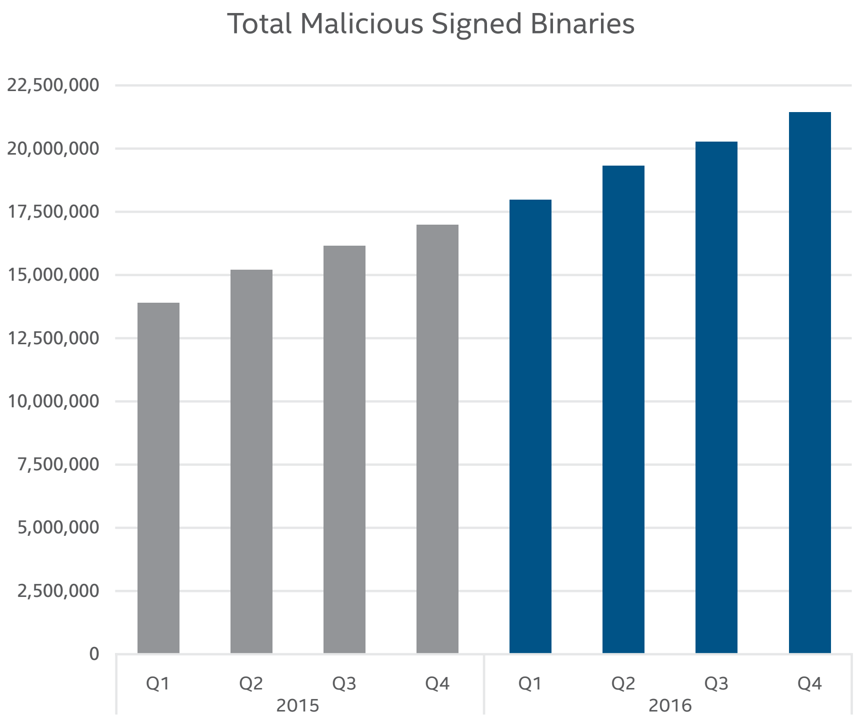


Source: McAfee Labs, 2017.

Share this Report



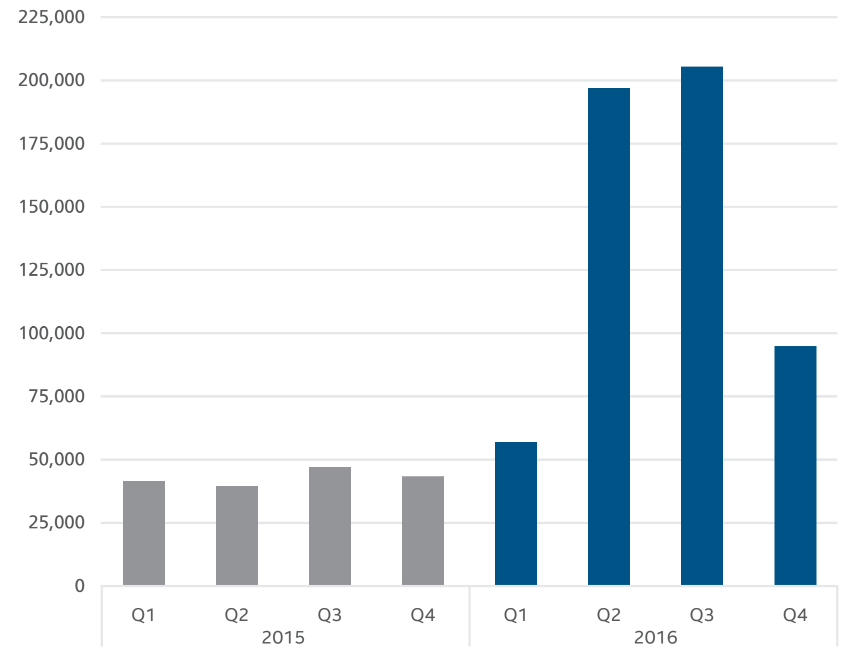
Source: McAfee Labs, 2017.



Source: McAfee Labs, 2017.

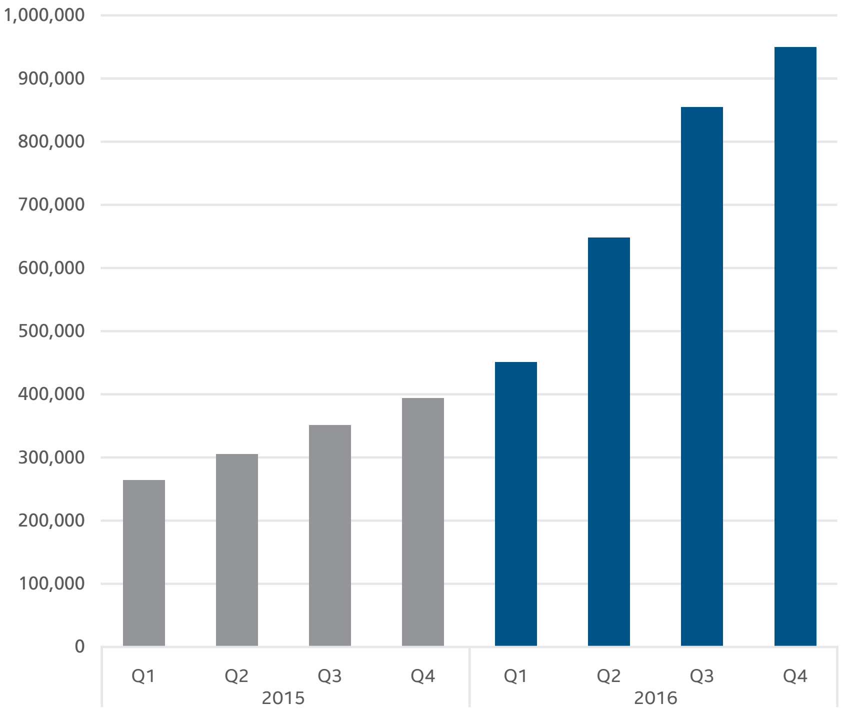
Share this Report

New Macro Malware



Source: McAfee Labs, 2017.

Total Macro Malware



Source: McAfee Labs, 2017.

Incidents

Starting with this threats report, we will include a set of charts related to publicly reported incidents. We gather incident data from public records, our threat research team, and Intel Security's Foundstone Services incident response team.

We report on incidents: security events that compromise the integrity, confidentiality, or availability of information assets. Some, but not all, of these incidents are breaches. Breaches are incidents that result in the confirmed disclosure (not just potential exposure) of data to unauthorized parties.

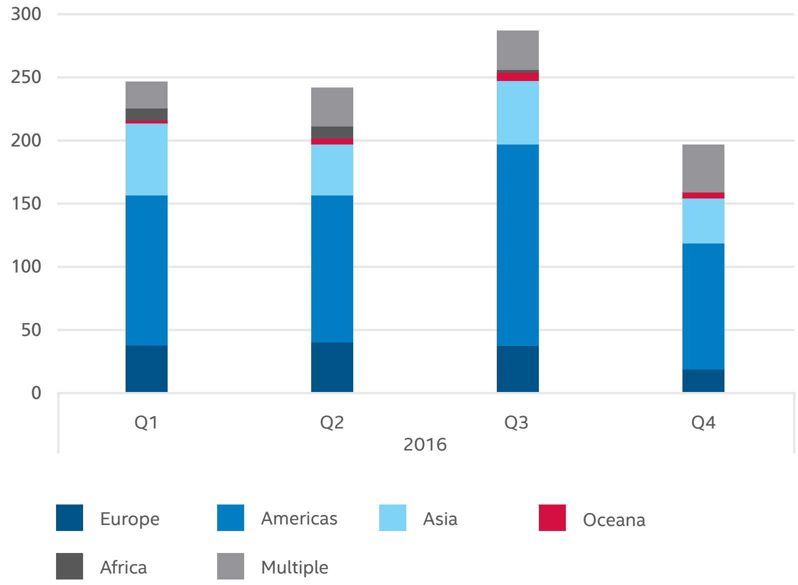
We counted 197 known public incidents in Q4 and 974 known public incidents in 2016. Organizations in the Americas experienced the majority, 493, of those incidents during the year.

The public sector experienced the greatest number of incidents by far. This may be the result of stricter requirements for reporting and the U.S. election in November.

A Q3 jump in incidents in the software development sector was due to the rise in attacks on gaming platforms. In the public sector, we saw a rise in attacks related to the U.S. election process, mostly voter database incidents and defacing. Outside the United States, the dispute between China and the Philippines regarding the South China Sea provoked attacks against both sides. In the finance sector, the SWIFT attacks led to a Q2 jump in incidents.

Known Public Incidents in 2016

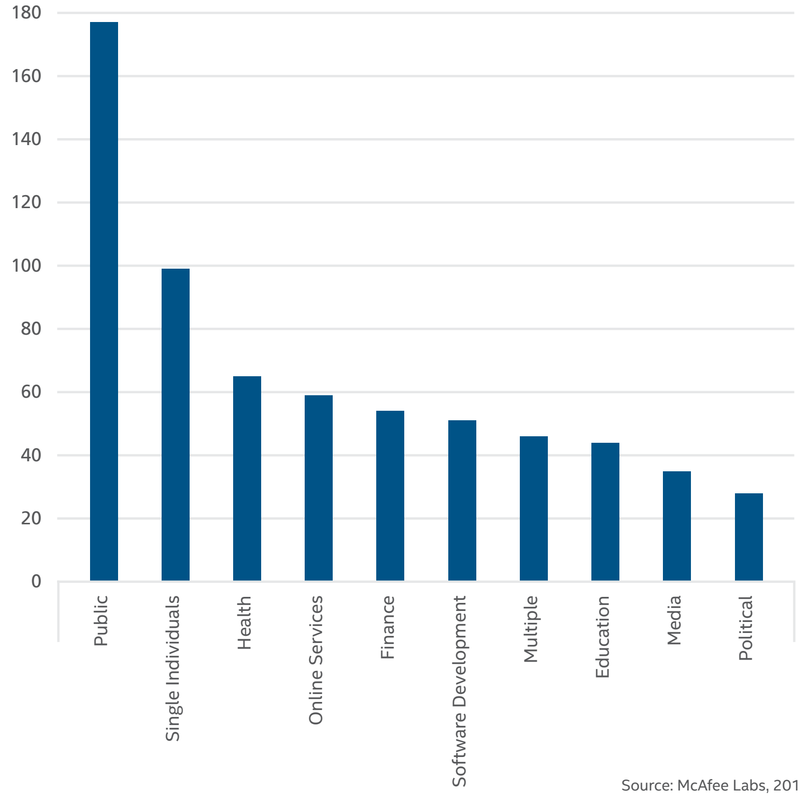
(number of incidents)



Source: McAfee Labs, 2017.

Top 10 Targeted Sectors in 2016

(number of incidents)

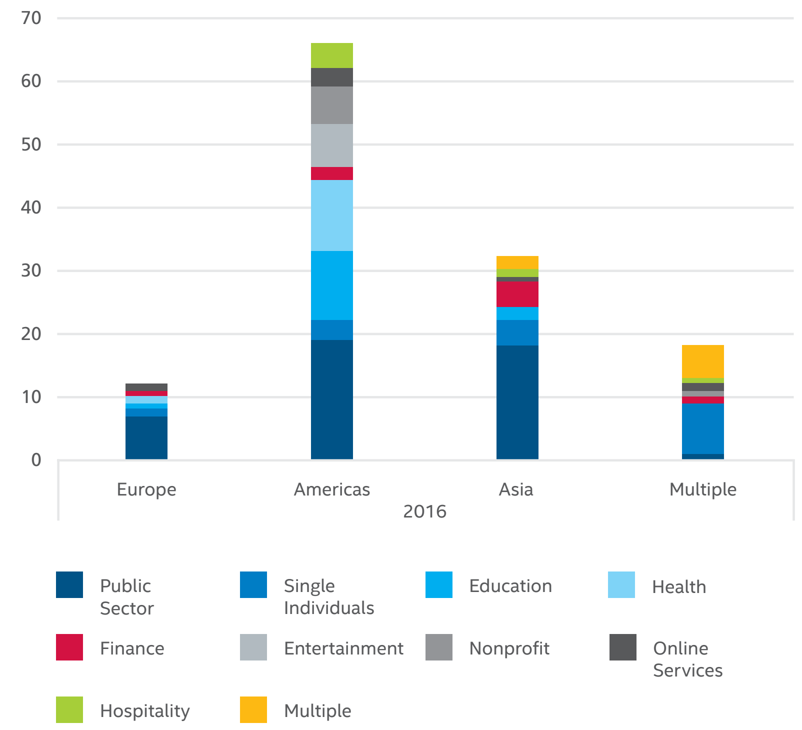


Source: McAfee Labs, 2017.

Share this Report

Top 10 Sectors Targeted by Region in Q4 2016

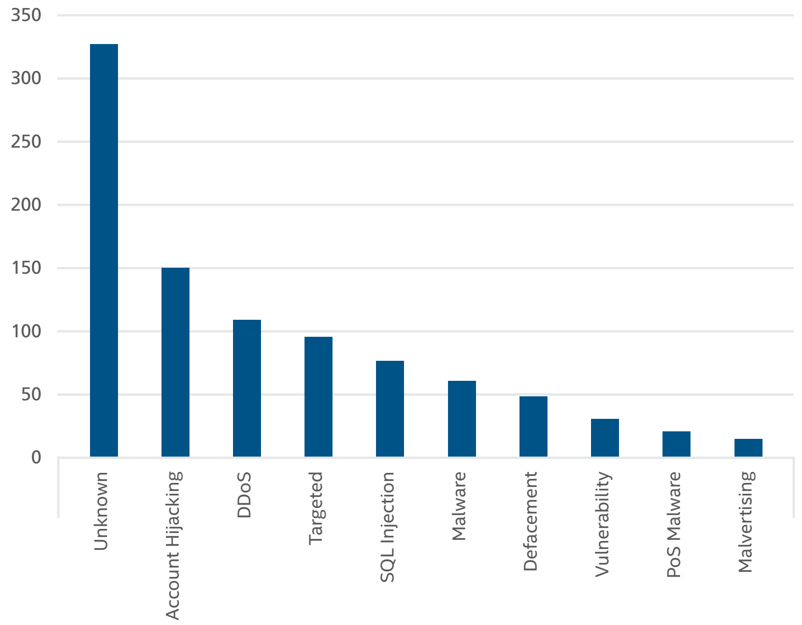
(number of incidents)



Source: McAfee Labs, 2017.

Top 10 Attack Vectors in 2016

(number of incidents)

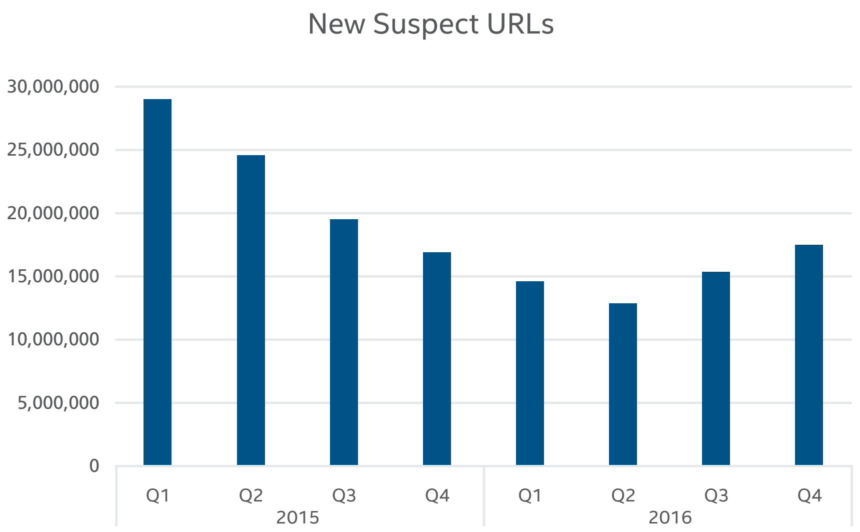


Source: McAfee Labs, 2017.

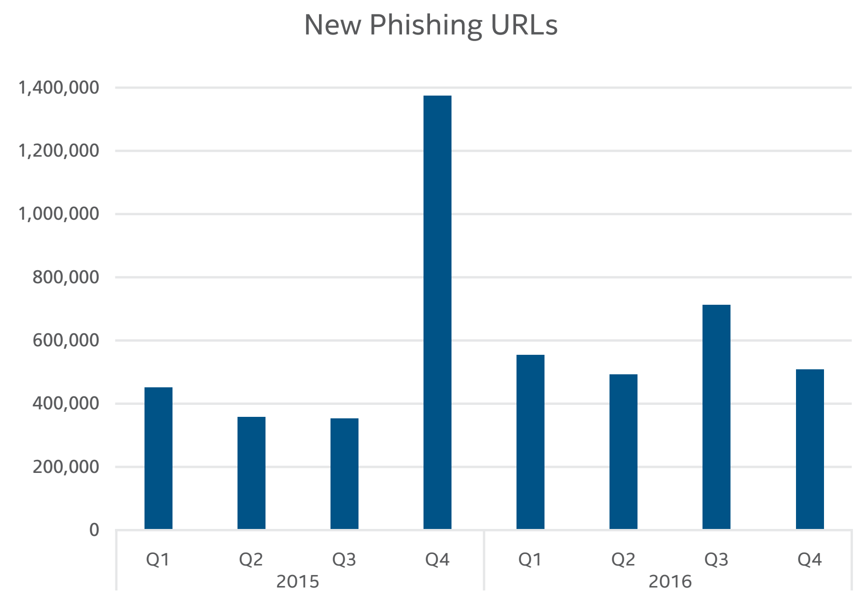
When an incident is publicly discussed, the attack vector is often not identified. That is why “Unknown” is the most common attack vector in this chart. This may be due to reporting requirements, the sensitive nature of the attacks, or other causes.

Share this Report

Web Threats



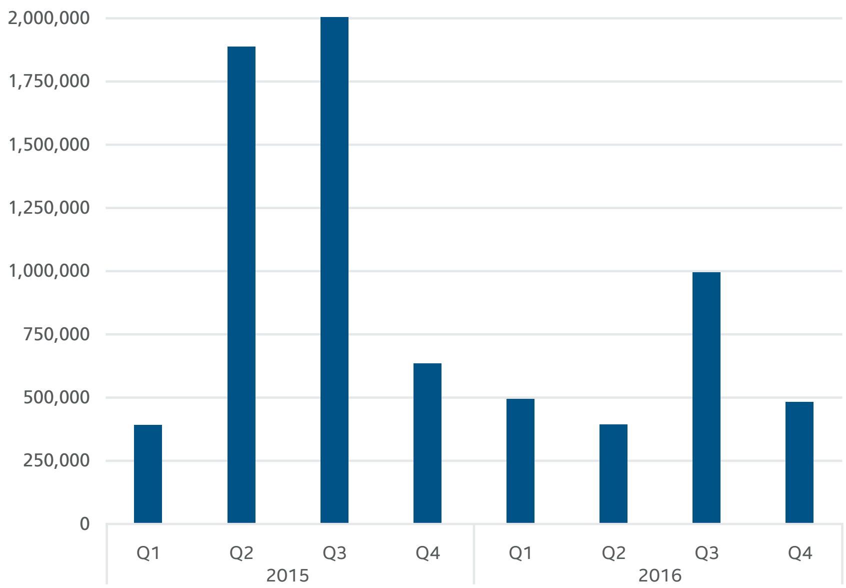
Source: McAfee Labs, 2017.



Source: McAfee Labs, 2017.

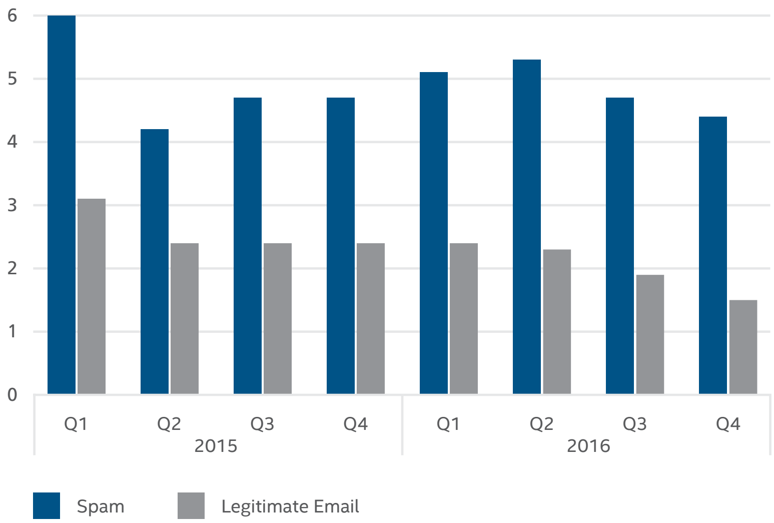
Share this Report

New Spam URLs



Source: McAfee Labs, 2017.

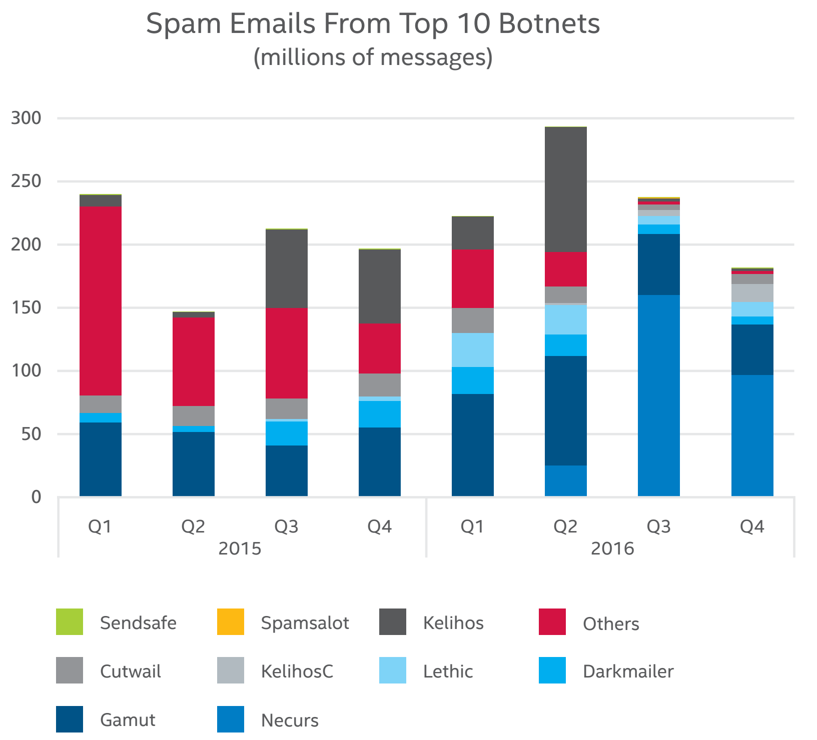
Global Spam and Email Volume
(trillions of messages)



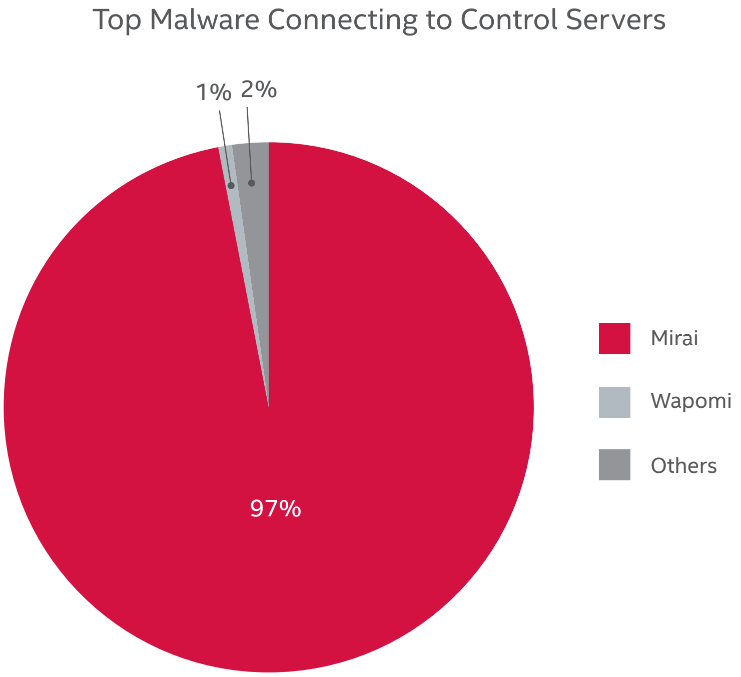
Source: McAfee Labs, 2017.

Share this Report

The KelihosC botnet, a recent purveyor of phony pharmaceuticals and Russian automotive supplies (such as "Winter and summer tires at competitive prices"), increased its overall volume during Q4.



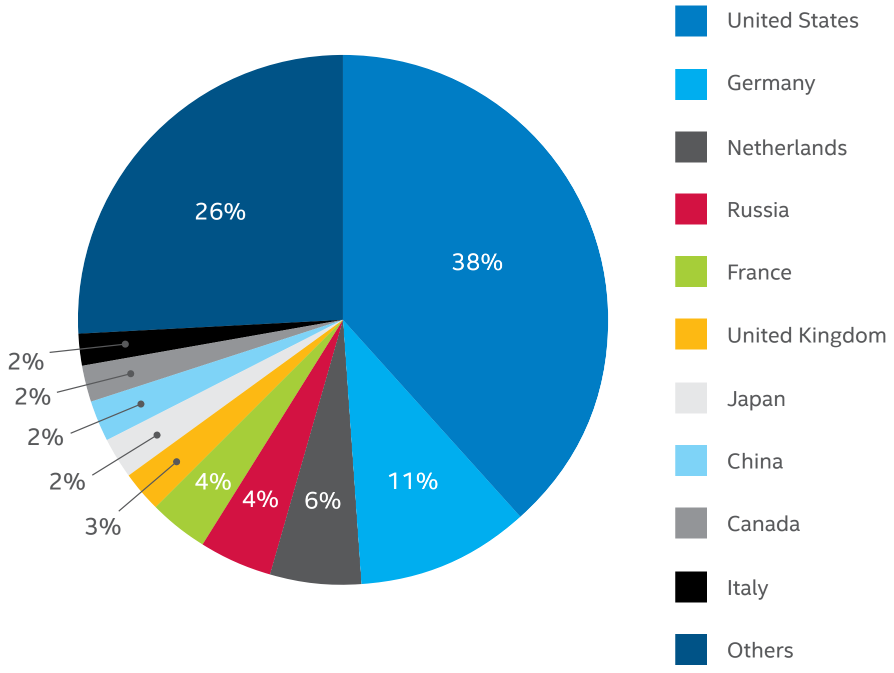
Source: McAfee Labs, 2017.



Source: McAfee Labs, 2017.

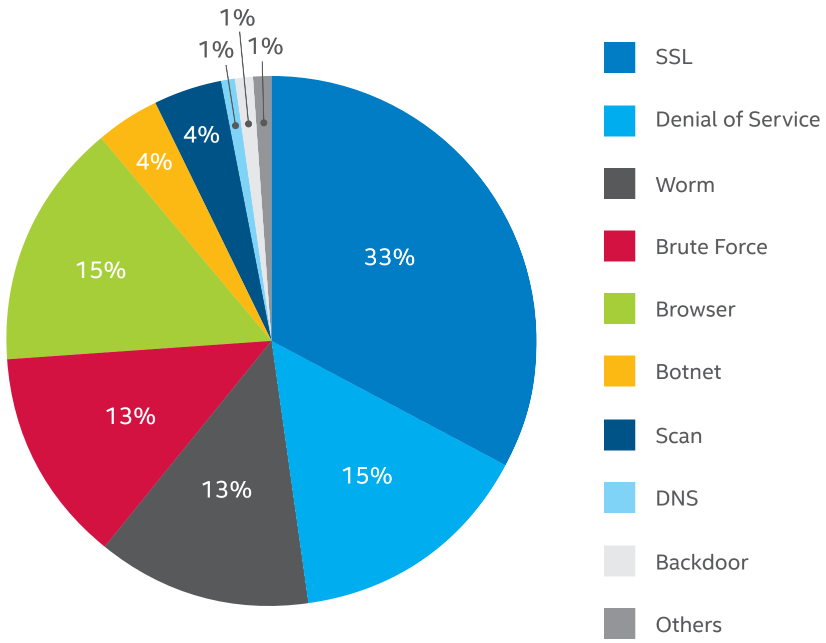
Share this Report

Top Countries Hosting Botnet Control Servers



Source: McAfee Labs, 2017.

Top Network Attacks



Source: McAfee Labs, 2017.

Feedback. To help guide our future work, we're interested in your feedback. If you would like to share your views, please click here to complete a quick, five-minute Threats Report survey.

Follow McAfee Labs

About Intel Security

McAfee is now part of Intel Security. Delivering proactive and proven security solutions and services that help secure systems and networks around the world, Intel Security protects consumers and businesses of all sizes from the latest malware and emerging online threats. Our solutions are designed to work together, integrating antimalware, antispyware, and antivirus software with security management features that deliver unsurpassed real-time visibility and analytics, reduce risk, ensure compliance, improve Internet security, and help businesses achieve operational efficiencies.

www.intelsecurity.com

McAfee. Part of Intel Security.
2821 Mission College Boulevard
Santa Clara, CA 95054
888 847 8766
www.intelsecurity.com

The information in this document is provided only for educational purposes and for the convenience of Intel Security customers. The information contained herein is subject to change without notice, and is provided "as is," without guarantee or warranty as to the accuracy or applicability of the information to any specific situation or circumstance. Intel and the Intel and McAfee logos are trademarks of Intel Corporation or McAfee, Inc. in the US and/or other countries. Other marks and brands may be claimed as the property of others.
2473_0417
April 2017