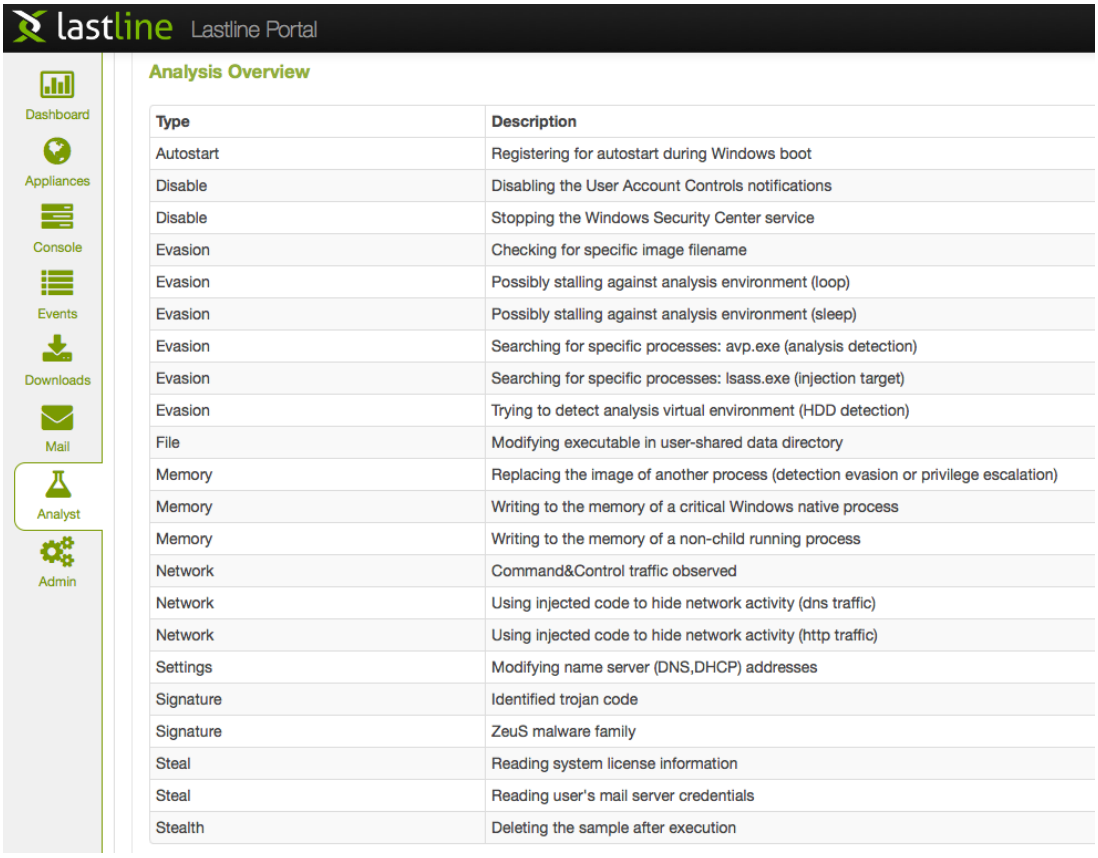
How Lastline Has Better Breach Detection Capabilities

By David Strom
December 2014
david@strom.com

The Internet is a nasty place, and getting nastier. Current breach detection products using traditional anti-malware sandbox technologies can't keep up with advanced persistent and hyper-evasive threats that pummel enterprise networks on an hourly basis. Malware authors encode their exploits with a number of operational vectors, so in case one entry point doesn't work they can still find a way into your network to do their dirty work. And as more businesses hire more outsourced consultants, part-time workers, and employ mobile devices, they open up additional mechanisms for malware to enter their corporate networks.

Some traditional AV and endpoint protection vendors have responded to these threats by adding features to their security products to do a better job of anticipating badly behaving packets coming through their detectors. They make use of limited virtual machines or operating system emulators to view how a piece of malware operates. That is great, but it isn't enough. Many malware authors can detect when these simulated environments are active and can evade detection accordingly. For example, some exploits such as *W32.DelfInj* can literally go to sleep for several days to avoid any detector that will just scan an infected system for the first several minutes. Here is a summary of the behavior of the Zeus Trojan, showing all the various evasive methods it takes to try to stay one step ahead of the detectors.



Type	Description
Autostart	Registering for autostart during Windows boot
Disable	Disabling the User Account Controls notifications
Disable	Stopping the Windows Security Center service
Evasion	Checking for specific image filename
Evasion	Possibly stalling against analysis environment (loop)
Evasion	Possibly stalling against analysis environment (sleep)
Evasion	Searching for specific processes: avp.exe (analysis detection)
Evasion	Searching for specific processes: lsass.exe (injection target)
Evasion	Trying to detect analysis virtual environment (HDD detection)
File	Modifying executable in user-shared data directory
Memory	Replacing the image of another process (detection evasion or privilege escalation)
Memory	Writing to the memory of a critical Windows native process
Memory	Writing to the memory of a non-child running process
Network	Command&Control traffic observed
Network	Using injected code to hide network activity (dns traffic)
Network	Using injected code to hide network activity (http traffic)
Settings	Modifying name server (DNS,DHCP) addresses
Signature	Identified trojan code
Signature	ZeuS malware family
Steal	Reading system license information
Steal	Reading user's mail server credentials
Stealth	Deleting the sample after execution

Fig. 1. How many different evasive techniques the Zeus Trojan uses.

This is the biggest fear that many security professionals have: that their tools might miss some advanced threat. All it takes for some of these exploits to happen is for a single to make its way through a next-generation firewall or APT security appliance (using a virtual or emulation sandbox that offers limited visibility) to someone's hard drive. For example, many "fake AV" malware exploits operate by sending a DNS request to a special command and control server that kicks off their infections to an endpoint device.

What is needed is a **next-generation sandbox** that can completely mimic a full-system (OS, CPU, peripherals) and glean intelligence from what happens when a particular piece of malware does its dirty business. This means being able to step through the execution code of a piece of malware and see exactly what it is doing to a host system: what system calls it makes, what registry resources it corrupts, what files and registry keys were modified and what payloads and pieces of remote control software it leaves behind. And which protective measures it avoids through various obfuscating measures. Some of the traditional end-point and network vendors are beginning to add a sandboxing feature as part of their detection tools, but lack sufficient comprehensiveness, scale and detection response time.

What is also needed is a way to **correlate a series of particular breach events** with the actual attack chain onset related to a malware infection, so that enterprises can remove these villains and quickly remediate their networks. Today's modern malware uses a combination of email, Web attacks, DNS redirects and mobile apps to work their way into a network. Any product should make use of both network traffic patterns and detect particular code objects from the network being delivered to endpoints. With the right kind of incident correlation, a security professional can examine what happened, where it happened, and the events that occurred before and after the actual infection.

Part of this analysis is to also add **IP and object based reputation analysis** to the mix, looking at known advanced threats associated with command and control servers and evasive malware.

And what is needed is to **do this in near real-time**, so that a potential exploit can be nipped in the bud, before it has a chance to spread across more endpoints and infect more machines. As the number of zero-day attacks and custom-created viruses continues to increase, the warning times get shorter. A number of security tools are now available that use distributed sensors to aid in their real-time warning systems.

So while there are tools that use these four criteria, few have combined them in the way that the Lastline Breach Detection Platform does. We tested it on a sample network that had been seeded with a series of malware infections. While no test bed can totally simulate the real world, we were impressed with the level of detail and its ease of use and the way it combined sandboxing, event correlation, IP/object reputation and real-time analysis. How does it work and how does it measure up to the above items?

Their core idea is to run a piece of suspected malware in such a way as to provide the ultimate examination of its operations. Suspected code is extracted from the network traffic flow, analyzed and correlated with other network-level events to provide a full picture of what happened. It has one of the most throughout analysis sandbox engines. But what is more important is how they are able to provide actionable intelligence to a wide variety of leading security vendors' intrusion prevention and unified threat management platforms from WatchGuard, Barracuda, TippingPoint, Juniper, Tripwire and others. Through a combination of application programming interfaces, Lastline can send and receive firewall blocking rules and breach event data to/from the appropriate systems that you have already purchased, so that these threats can be quickly stopped.

Lastline has four major components:

- **Network sensors.** This is software that can be installed on standard servers or VMs that continuously monitor network traffic through switch span ports to collect suspicious behavior. Lots of security tools do this already, and certainly this is the cornerstone of any modern security tool. What makes Lastline more interesting is that it combines IP and domain reputation analysis with malware fingerprinting techniques. With its 6.0 release the Lastline breach detection platform now includes unlimited 10 Gbps sensor interfaces, too.
- **Advanced sandbox screening tool**. Suspicious objects that are suspected to be zero-day threats are collected from the sensors and analyzed with the Lastline next-generation sandbox, which emulates a complete endpoint system (OS, memory, and peripherals). Other sandboxing tools leave small in-guest code stubs that can reveal they aren't "real" endpoints; Lastline doesn't have these clues for malware to key into and looks just like regular computers. In addition to running the code, this tool also records the specific attack chain behaviors of the malware code and documents what harm it is actually doing to the system, including delivery, detailed exploit characteristics, malware installation and command and control communication.
- **Reporting and threat analysis tool.** Low-level event data is then collected and correlated into a particular security incident, which then updates an online threat database. You can examine the captured network traffic and malware behaviors, see the results of the analysis, and look at why this was suspicious or threatening with low false positive/negative noise. Information is displayed on a highly graphical and easily parsed Web-based portal for action by security administrators. For example, just by clicking on a few different menu items, we can see how often the same infection was downloaded by a particular endpoint, or why a particular event led to other activities across our network, or how a piece of malware was attached to a series of different email messages.
- **Rich threat intelligence of advanced threats.** Known exploits and IP based systems associated with advanced malware are highly dynamic and

traditional signature-based knowledge bases are ill equipped to keep up. Lastline threat intelligence draws on its global collection of next-generation sandboxes.

To add flexibility to its system, both the next-generation sandbox and reporting tool can be either hosted or installed on-premises. Here is an example screenshot of the reporting tool, showing a summary of incidents, the top 10 malware infections observed, and a listing of each event that you can drill down for further analysis.

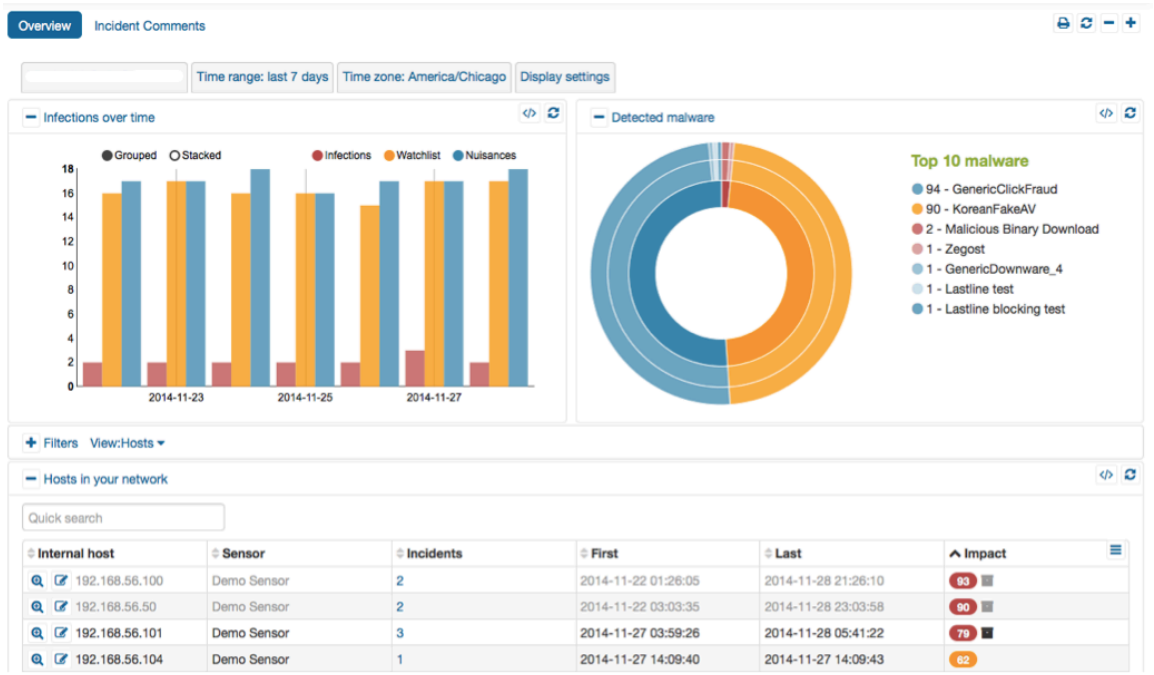


Fig 2. A typical dashboard from Lastline's platform, showing particular incidents and infections.

Summary

Other security tools have some but not all of the components that are found in the Lastline Breach Detection Platform. What makes them unique is their range of discovery, the way they can effectively mimic actual PC or smartphone endpoints to examine malware behavior, and how they can scale up to handle very large networks with their modular and SaaS-based tools. The Lastline platform offers a predictable annual subscription model starting at $40/user/year, offering enterprises the ability to monitor unlimited number protocols and locations as well as analyze 10Mb to 10G networks without incurring any additional fees.

About David Strom

David Strom (@dstrom, strominator.com) is one of the **leading experts on network and Internet technologies** and has written and spoken extensively on

topics such as VOIP, convergence, email, cloud computing, network security, Internet applications, wireless and Web services **for more than 25 years.** He has had several editorial management positions for both print and online properties in the enthusiast, gaming, IT, network, channel, and electronics industries, including the editor-in-chief of *Network Computing* print, DigitalLanding.com, and Tom's Hardware.com. He began his career working in varying roles in **end-user computing in the IT industry**. He has a Masters of Science, Operations Research degree from **Stanford University**, and a BS from **Union College**.