BINARY DEFENSE™ | 2021 White Paper

ABNORMAL BEHAVIOR DETECTION IN THE ENTERPRISE

A HACKER'S VIEW OF ENTERPRISE SECURITY

By David Kennedy
Binary Defense Chief Technology Officer

Most organizations struggle with the ability to detect attack vectors that are designed to evade enterprise defenses. The shifting tactics of the attackers are troublesome for most companies due to the nature of how the attacks work and the inability to change dynamically with the attack vectors. Traditional technology such as antivirus, Intrusion Prevention Systems and firewalls are a base level of security when it comes to defending against what most attackers focus on. With the latest attacks we've seen in supply chains, the attacks continue to get more sophisticated and attacking our third party trusted organizations to deploy attacks against our own infrastructure.

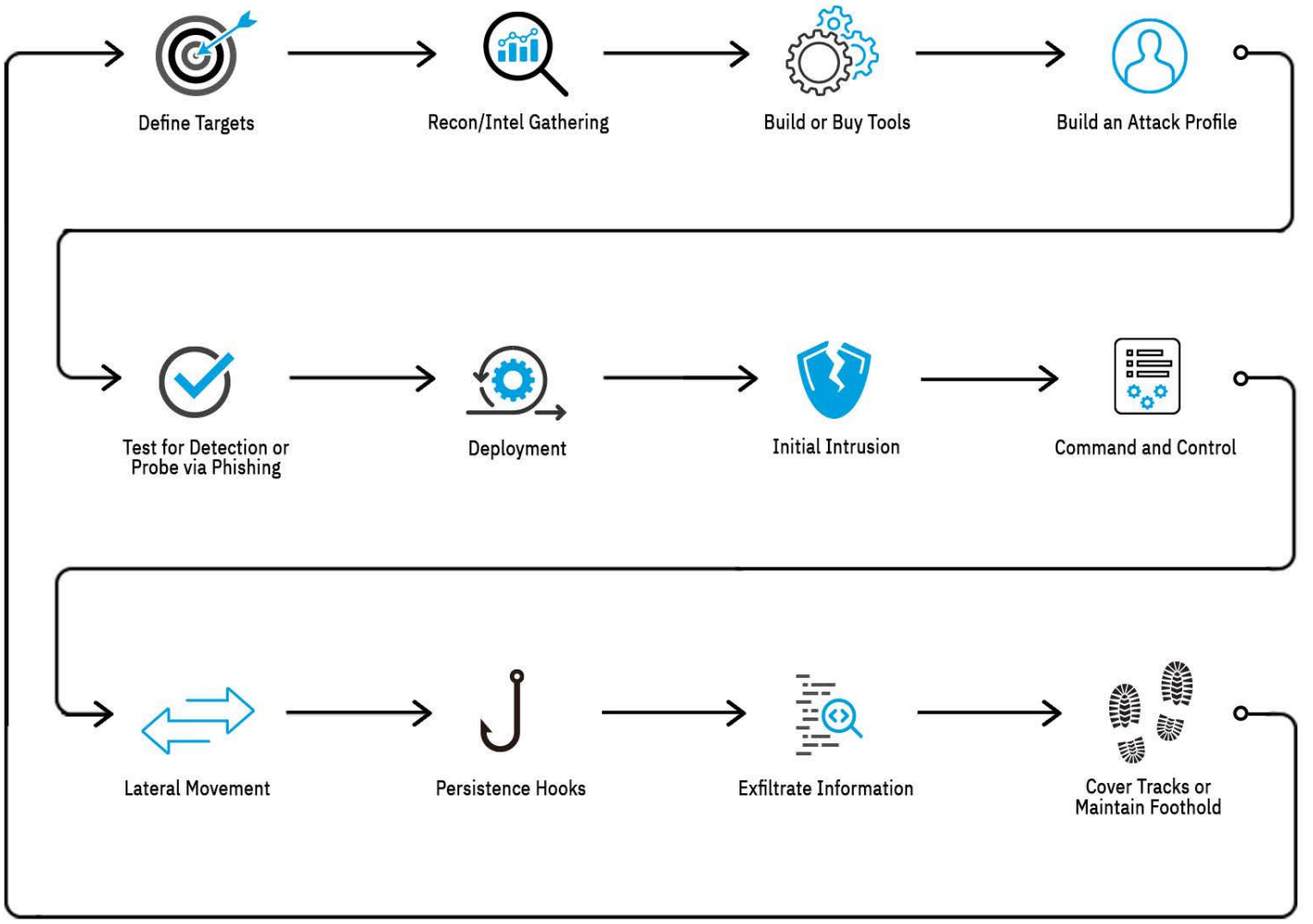
"Most organizations struggle with the ability to baseline their images and ensure a consistent number of workstations / endpoints that have the same configuration across the enterprise."

As an attacker myself, I focus my efforts on ensuring that when I go after a target, the implants (malware/backdoors) do not get detected by antivirus and other technologies. Some companies have invested in sandboxing technologies that focus on the virtualization and identification of abnormal patterns. These are highly predictable and easy to circumvent. To this day, when looking at breaches on Microsoft-based systems, often times executables/binaries and PowerShell are some of the most common attack methods. That means that most of the attack vectors, regardless if it's a macro-enabled Excel document or visiting a malicious website, attempt code injection to download an executable or execute PowerShell in order to compromise the system. This comprises the majority of attacks that we see today in the industry.

Executable infections are problematic because they range from ransomware infections all the way to targeted attacks from nation states. With the majority of the "noise" being from executables, it's never been more important for us as an industry to understand the best approach to protecting against the largest risk factor we have. While our endpoints tend to be the infection origination for a breach, the methods for infection need to be understood and protected against. The question is, how do you protect yourself against the largest population of attacks, and then move onto the more advanced methods for compromise.

This brings us to the topic of "known good" or what is also known as allow-listing. Most organizations struggle with the ability to baseline their images and ensure a consistent number of workstations/endpoints that have the same configuration across the enterprise. The concept of known good takes a company's normal operations and documents deviations in order to understand what the enterprise needs to operate. Any deviations are then either prohibited or monitored in order to ensure they are not malicious in nature. For those not familiar with allow-listing, it's the concept of baselining your organization and then from there only allowing what is normal. "Normal" is defined by the baseline configurations and documented deviations and monitored from there.

Lifecycle of an Attack



The concept of "known good" is nothing new, but if your main infection method is done through executables and you understand what your environment is doing, you can eliminate 90% of your "noise" which is the main method for exploitation and then monitor on deviations. The cybersecurity industry is focused too much on individual attacks and not on how to reduce the overall noise of an organization and focus/prioritize on the best methods for reduction of noise and minimization of risk.

There's no question that application allow-listing is difficult. It's a GOOD difficult. It means that when you baseline your enterprise, you have an understanding of what your environment looks like. It means that you can now look for deviations of patterns and recognition of behavior. Most security programs are not anywhere near this level. Once you've performed and implemented known good, it becomes significantly easier to prohibit directed attacks against you.

There's no question that application allow-listing is difficult.

It's a GOOD difficult. It means that when you baseline your enterprise, you have an understanding of what your environment looks like. It means that you can now look for deviations of patterns and recognition of behavior. Most security programs are not anywhere near this level. Once you've performed and implemented known good, it becomes significantly easier to prohibit directed attacks against you.

Here's an example of how to configure known good. Let's take a basic example. Most attackers (98.7 percent according to Binary Defense research) do not utilize code signing certificates. If you implement a program that blocks any executable that is not code signed (especially in user profile directories), you can eliminate 98.7 (on average) of risk in your environment.

Example:
Microsoft deploys a patch. Is code signed by Microsoft. Allowed. Malware - not code-signed, is blocked.

In this simplistic example, if you block anything that is not code-signed and allow exceptions based on deviations, known good becomes a much easier process to handle.

I am personally a huge advocate, but I'm not alone. Penetration Testers/Researchers/Red Teamers all agree that by baselining your configuration you can drastically reduce your attack surface.

Let's assume that you have bought into the concept of application allow-listing and "known good" and have it implemented appropriately. Now comes the detection and deviation of patterns. Attackers are smart. They realize that organizations that have implemented known good will become much more difficult for exploitation. They need another way for to exploit.

Multiple other methods to gain access to a system don't require exploitation of executables. PowerShell is a fantastic example. There are patterns within an enterprise that you can baseline, similar to known good that can help you detect these types of attacks.

Did you know that there are fourteen different variations to EncodedCommand which are used for PowerShell detection bypasses?

```
-e
-ec
-en
-enc
-enco
-encod
-encode
-encoded
-encodedc
-encoded co
-encodeecom
-encodedcomm
-enodedcomman
-encodedcommand
```

There's great research on PowerShell injection as a main method for exploitation by Palo Alto Networks on methods for exploitation using PowerShell:

http://researchcenter.paloaltonetworks.com/2017/03/unit42-pulling-back-the-curtains-on-encodedcommand-powershell-attacks/

Binary Defense focuses on "known good" as well as enhanced detection and prevention capabilities around obscure behavioral attack vectors. While known good may seem daunting, our proprietary Managed Detection and Response solution focuses on known good protection (in an easy manner) as well as multiple other phases of an attack. All the way from methods of compromise, to lateral movement and further compromise of systems.

Our detection is truly best-in-class. We were recently recognized in the "Leader" category in the Forrester WaveTM Managed Detection and Response Q1 2021 report. The report stated, "Security buyers looking for a rapidly growing MDR focused provider with a clear emphasis on security research and threat detection should evaluate Binary Defense."

The report continued with stating that "Collaboration and partnership stand out as key elements behind its service delivery to ensure that security practitioners have what they need to detect, investigate, and response to security incidents."

About David Kennedy

Dave Kennedy is on a mission to advance the cybersecurity industry on a global scale. As Binary Defense Co-Founder and Chief Technology Officer, his experience and expertise guide our cybersecurity solutions, including the development of our proprietary managed detection and response technology.

Named one of the Top 10 IT Security Influencers in the World by CISO Platform, Dave has more than 15 years of experience in the security industry.

Prior to forming Binary Defense, Dave founded TrustedSec, an information security consulting company located in Northeast Ohio, which specializes in attack simulations with a focus on strategic risk-management.

In an effort to advance the industry, David co-authored 'Metasploit: The Penetration Testers Guide' and co-founded the 'Penetration Testing Execution Standard' (PTES), which has been adopted by the Payment Card Industry (PCI). David is the creator of several popular open-source tools, including 'The Social-Engineer Toolkit' (SET), PenTesters Framework (PTF), Artillery, and Fast-Track. In addition to focusing on research, Dave has released a number of security advisories, including zero-days.

He has appeared as a guest on multiple news networks including Fox News, CNN and MSNBC. His tools have been featured on the History Channel and the popular USA Network television series Mr. Robot, where he also assisted with content.

600 ALPHA PARKWAY, STOW OH 44224 | 800-BINARY2

WWW.BINARYDEFENSE.COM