SANS

# Six Defensive Techniques to Make Your Attackers Cry:

## *Russia and Ukraine Cyber Crisis*

### You can win on the defensive side of InfoSec. We’ll show you how.

In this paper, there are six incredibly effective defensive techniques. They will work for organizations of all sizes.

The goal of these controls, isn’t just to stop attackers, but rather to create a positive feedback loop. If you follow these steps, you will reduce your noise, which allows you to do more meaningful work, which reduces the noise further, and so on!

If you’ve ever felt trapped on an IT/cybersecurity treadmill, this is your escape plan. This is a blueprint for victory as a defender.

One key element: You should NOT immediately be thinking of purchasing security solutions.

**Your organization’s procurement process is likely too long.**

**Due to current supply chain issues, delivery of products is taking weeks (or more).**

**Newly purchased solutions take time to deploy and, in many cases, even more time to integrate into your security stack.**

Because of these limitations around securing new tooling, we’re going to focus on leveraging your existing tooling to their maximum extent. In short, we’re going to get creative.

**Your path to winning comprises of six techniques:**

1. Patch
2. Execute tactical and effective logging strategies
3. Control outbound traffic (and geo blocking)
4. Plan and test for rapid containment
5. Implement application control
6. Make this a sustainable “steady state”

**For each of the above items, we will show you:**

- Why you need it
- How to do it (with a focus on using existing systems or lower-cost solutions)
- Tips to avoid common mistakes

# Using This Document

Protecting your data requires careful and constant stewardship. In this document we will show key strategies to radically reduce the level of effort required at the tactical level.

This resource isn't meant to be a radical departure from how IT or infosec are done. However, for some organizations, these methods outlined will be a dramatic shift. The authors of this paper strongly suggest that you quickly review all six steps outlined below. If your org is not performing these tasks, you likely have a far greater exposure to attackers than you realize.

We strongly recommend you approach this list in the order it's presented. They're listed in the order of impact. However, if one of the items is too difficult or problematic, skip it. We'd rather you skip multiple steps and be met with some level of success, than get blocked at any single phase.

Unlike other roadmaps, the tasks outlined here are meant to be part of an iterative improvement process. We hope you will run through this playbook multiple times. To make these future reviews easier, we highlight potential next steps to take when you revisit these tasks.

# 1 Patch

## WHY YOU NEED IT

The Cybersecurity & Infrastructure Security Agency (CISA) published 383 vulnerabilities to the Known Exploited Vulnerabilities Catalog. These vulnerabilities are regularly used by multiple threat actors to gain initial access, escalate permissions, or move laterally within a network. Applying the appropriate patches to your systems will complicate these three activities. Of the 383 vulnerabilities, *382 can be remediated by patching!* The one outlier? It's for an end-of-life product, so no patch is available.

## HOW TO DO IT

First, prioritize your patching efforts by ease of exploitation by a remote attacker. Broadly speaking, this means patch in the following order:

### Internet Accessible System and Software

- Network and Security Appliances: firewalls, VPN concentrators, load balancers, proxies, etc.
- Web servers (Apache, Nginx, IIS, etc.), web applications, mail appliances, mail servers, FTP servers, and SSH/SFTP
- Web Applications
- Host OS for Internet Accessible Systems

### Clients and Client-Side Software

- Microsoft Office
- Adobe PDF products
- Browsers, in order of popularity (Chrome, Safari, Edge/Internet Explorer, Firefox)
- VPN Clients
- Mobile Devices and Software
- OS Updates

### Internal Servers and Server Software

- Windows Servers
- Internal Web Applications
- IoT Devices

Utilize the automatic updates features of the software and systems in your environment. Ensure those update mechanisms are configured on the operating system and key software components.

Microsoft update:

- Windows 10 and later automatically downloads and installs patches from Microsoft Update.
- If you have deployed custom Group Policy settings for Windows Update, ensure the settings do not require users to manually install updates. The following link describes how to Allow Automatic Update Immediate Installation: Configure Automatic Updates by Using Group Policy | Microsoft Docs

Second, scan your network for the CISA Known Exploited Vulnerabilities. Scanning for these vulnerabilities can identify systems that have slipped through your patching process, or are vulnerable because of misconfiguration.

Scanning for vulnerabilities:

If you are a Federal, state, local, tribal, territorial governments, or a public and private sector critical infrastructure organization, CISA will provide you with Cyber Hygiene Services at no cost, which include vulnerability scanning and web application scanning.

If you are a Qualys customer, check out the following dashboard to focus on the CISA Known Exploited Vulnerabilities. If you are a Tenable.SC customer, there is a preconfigured dashboard as well.

## TIPS TO AVOID COMMON MISTAKES

### The "I MUST Patch It All" Trap

Sometimes, you just cannot patch a system. Or maybe a patch will take too long to fix. If you're in a crisis state, it sounds odd to say this, but you should NOT spend too much time trying to force a patch. Instead, isolate that system through network or host-based firewalls to limit the exposure to the rest of your network. Many security programs fall into "we must patch it all before we move to the next item" fallacy. That is almost never a good idea. You should patch what you can if at all possible, but if you cannot patch something... isolate it, and move on!

### Analysis Paralysis

Testing patches can be a full-time job. If you're applying patches from the software's publisher, you likely will be ok to just deploy the patch in a non-production instance, or a lesser critical system to verify nothing is broken.

If you're especially concerned about patches breaking things, consider doing a phased rollout after a brief non-prod test. Many organizations take a phased approach where they will apply patches to a few (handful of test systems), some (10%), many (30%) , and finally any remaining systems. Save your most sensitive systems for the end of the patch rollout. If you feel like you're spending too much time and effort on patching, you probably are.

### Patch the Low and Medium Issues Too

Understanding vulnerabilities is difficult. Sometimes multiple "Low" or "Medium" severity vulnerabilities can be chained together and result in an aggregate risk much higher than the individual risk of each vulnerability. Using vulnerability severity to patch only "the most critical" vulnerabilities may leave you exposed to an aggregate risk that is dramatically higher than you would expect.

### Patch Beyond the OS

Ensure you are patching unmanaged client software, too, such as third-party browsers, Adobe products, etc. These clients are common targets for attackers and often go unpatched for years. The current model for many client applications is to notify users of available updates, but not force the installation. Don't let these applications slip through the cracks.

### ▶▶▶ POTENTIAL NEXT STEP

**If patch validation is too time consuming, consider automating key parts of the process. Through a long-term lower level of effort, a library of unit tests can be built. These tests are quite handy for many reasons, but are especially powerful for patch validation.**

**Apache JMeter is a free tool that allows you to record just about any computer action. Why not use it to record a critical business transaction? Once you have the recording, you can radically speed up your patching.**

**When the next set of patches are released, use this workflow:**

- **In a non-production environment, verify the recorded transaction still works as expected.**
- **Then undo the transaction.**
- **Apply the patches.**
- **Run the recorded transaction.**
- **Verify the transaction worked as expected.**
- **If the transaction worked, your patch validation is done!**

### Aggressively Hunt for "Lost" Systems

Don't assume your inventory systems are accurate. Search for the systems that have slipped through the cracks and remain unmanaged. Vulnerability scans can help, but also consider broad discovery scans with Nmap, Masscan, or similar to identify unmanaged hosts.

Additionally, look in your logs for unexpected hosts. Your DHCP, DNS, and Active Directory logs are great sources of passively finding "lost" systems.

# 2 Execute Tactical and Effective Logging Strategies

## WHY YOU NEED IT

Without logging, you have limited visibility. Logs are your eyes and ears when it comes to everything computer based. Without logs, you will have almost no understanding of what's happening in your environment. Additionally, with appropriate logging, audits and other regulatory reviews will go faster and easier.

## HOW TO DO IT

If you are an organization that has email and office capabilities in the cloud, you should use the native logging that is already present. After all, you have paid for these functions in your subscription. If your organization uses M365, you should start with the built-in Security Center. If you use Google Workspaces, you can find security logging in the Admin Report Center.

For organizations who are still on prem, Windows Event Forwarding (WEF) is a native and powerful solution. It allows Windows systems to stream events (logs) to a Windows Event Collector. Microsoft provides guidance on how to use WEF to detect attacker behaviors in your environment.

## TIPS TO AVOID COMMON MISTAKES

### Don't Rely on Defaults

Sometimes we put too much faith in our vendors. Chances are, they have not been in your environment. They have not worked directly with you and your team. Because of this, default logging levels are their best guess and likely do not reflect your actual logging needs. Who knows best what you need? You do!

Instead of over relying on the vendor, you should ask, "what stories do I want to tell?" If you want to catch attackers trying to brute force logins, you will need to know when a login failure happens. Taking a story-based approach is typically called Use Case Based logging.

### Logging is an Ongoing Process

As you progress on your logging journey, you will learn different use cases and logs you'll want to gather and analyze. This means that your logging approach will change over time. You should plan and set expectations that logging is an iterative project. Organizations who successfully do logging, will often take a phased approach.

Phase 1: First start with a guided site like what2log.com (Which provides tips on what you should log and how to do it)

Phase 2: Then go with something more strident. For Windows systems, Microsoft's Logging Guide is very detailed.

Phase 3: Finally, you could use a system like Sigma. Look through the various rules to see an alert that is appealing to you. By viewing the dependencies listed for that entry you will learn log sources you will need to collect.

### Avoid Log Hoarding

It's tempting to enable all logging options, but you'll quickly find yourself drowning in logs of little or no value. Not only does it become too much info to act upon, it can be very expensive in terms of log storage, network utilization, and any licensing fees for your logging systems. To combat these problems, we urge you to take a use case based approach. Pick a story you want to tell and then "walk backwards." What data will you need to tell this story?

### Treating All Logs Like They Are Regulated

Not all logs are created equal... at least in the eyes of the law. You may have regulatory or contractual requirements to retain logs for a specific time (perhaps even years). Many organizations shy away from logging entirely when they see these requirements because they feel it may be better to not log at all and avoid these retention issues.

Fortunately, there are no laws that require an organization to keep *all logs* for the retention time. In fact, many forward thinking orgs delete non regulatory protected logs once they are no longer needed. In some cases, you can safely remove a log after one week.

### ▶▶▶ POTENTIAL NEXT STEP

**To speed analysis, some organizations choose to collect their logs into log aggregators or a SIEM (Security Information and Event Management) solution. The selection, installation, and deployment of such a solution is a major undertaking, and something that *should not be done as part of your triage and crisis response process.* Please review the SANS Reading Room Paper "An Evaluator's Guide to NextGen SIEM" to get a better understanding of the order of magnitude of this work. If your organization already has a SIEM type solution in place, a review of this guide is still worthwhile. You might not be using your tooling to the fullest extent.**

**As a potential stopgap before getting a full SIEM tool, consider the use of event correlation and light touch analysis products like WEFC, which leverages WEF to help centralize analysis.**

## 3 Control Outbound Traffic (and Geo Blocking)

### WHY YOU NEED IT

Controlling and monitoring outbound network communications is one of the most effective ways to identify and disrupt the delivery of exploits, payloads, and command and control traffic. Attackers require network connectivity for most aspects of their attacks and restricting outbound traffic to only what is absolutely necessary complicates the attack process.

### HOW TO DO IT

There are countless ways to achieve this, but the main components are firewall rules, web content filters, DNS content filtering, and network monitoring tools. Let's look at each.

### Firewall Rules

Never underestimate the effectiveness of egress firewall rules. While they are deadly effective, they can also be complex to manage. It's reasonable to create blocklists based on known malicious IP addresses, or even geographic location based on Regional Internet Registries IP ranges. Joff Thyer's RIRTools makes this a snap, and is actively maintained.

### Web Content Filter

Most internet sites are categorized by various web content filtering companies. The feeds from those products are a great way to filter internet traffic based on known categories. This allows you to allow or block specific categories, but it also allows you to block access to uncategorized domains. Less sophisticated command and control traffic is done with recently registered domains, which likely have never been categorized. So, blocking uncategorized raises the bar. Web content filtering is normally done on "next gen" firewalls or through a web proxy. If you don't have a security appliance with that feature set, DNS content filtering is a decent alternative.

### DNS Content Filtering

The same "categorization" approach web content filters use is applied to some DNS servers. This allows the DNS server to return the site's actual IP address if the category is allowed, or the IP address of a web page that indicates the domain was blocked by the filtering service. The downside to this approach is it assumes the attacker is using DNS names for systems involved in the attack. If the attacker is using simple IP addresses, this filtering will be ineffective. It also requires that all systems in the environment use the appropriate DNS servers. So, it is essential to configure egress firewall rules to only allow DNS traffic to the DNS Content Filtering service. If possible, it's even better to only allow the trusted internal DNS servers to communicate with the external DNS systems, and force all internal network hosts to use your organization's internal DNS servers. Additionally, consider implementing Mark Baggett's freq.py, freq_server, and domain_stats.py utilities. These tools help identify malicious domains that are either algorithmically generated or recently created. He details using the tools in a talk at Security Onion Con, be sure to check out the latest version of the tools he mentions.

### Network Monitoring Tools

The goal of this paper is to provide you tools and architectures that you can deploy quickly if they aren't already in place. While strict egress filtering based on "Allow Lists" of specific sites, services, and IP addresses would ensure no system communicated with any untrusted internet hosts, it's a lofty goal that might not be attainable. As you work towards such a lofty goal, it's critical to have an effective network monitoring system in place to enhance your visibility to detect the attacker that has evaded the other filtering techniques we've discussed.

## TIPS TO AVOID COMMON MISTAKES

### Consider Open Source

In most areas of IT and Security, there are a mix of open-source and commercial solutions. In the network monitoring and analysis space, open-source solutions typically are more feature rich than comparable commercial offerings. If open-source solutions are problematic because your organization requires professional support, it is common for the developers of these applications to offer professional services, or partner with (and suggest) competent providers.

### Take a Modular Approach

As mentioned throughout this section, enhancing visibility and controls at the network layer can be a massive undertaking. Most network control projects fail due to a scope set too large. It is much easier to deploy a focused set of controls around a set of desktops used by high risk users or perhaps critical servers.

**▶▶▶ POTENTIAL NEXT STEP**

**Consider segmenting your network into functional zones to limit the ability of attackers to access resources once they gain access to a single system. If your network is already segmented, consider moving to a micro segmentation network topology.**

# 4 Plan and Test for Rapid Containment

## WHY YOU NEED IT

One thing that many defenders misunderstand is the speed of an attack. You should start planning for ways to disrupt your adversaries in real time. Actions such as taking systems, or even network segments, offline during an attack will prevent attackers from achieving their objectives.

## HOW TO DO IT

### Isolate Systems or Networks

Work with the system owners to determine what systems can be brought offline, and under what conditions. We strongly recommend you use this form (Pre-Authorization to Take System or Network Zone Offline. SEE APPENDIX. ▶)

When you have reason to believe a system has been compromised, you'll need to move quickly. Physically unplugging it from the network is typically the best. This allows the most evidence to be preserved and collected. Please work with your incident response provider to determine what evidence they'll want BEFORE an incident. You should specifically ask if they plan on doing memory analysis. If they will be doing this, it is vital that you not power down a system until they explicitly request this of you. Powering off a system clears this memory, denying them key evidence they will want to use.

### Disable Accounts or Password Reset

Attackers will frequently hijack valid user accounts. Be prepared to disable accounts or do password resets for multiple affected accounts quickly. Be ready to remove accounts from groups to which they do not belong.

## TIPS TO AVOID COMMON MISTAKES

### Practice

Many organizations underestimate the level of effort required to do these actions in a coordinated manner. Before attempting technical testing, first do "table top" events to verify the different groups that are needed to coordinate these activities to understand what roles they play in the process.

### Second Guessing vs Process Improvement

It's important to review performance to ensure work done meets expectations. It's unfair and counterproductive to apply after-the-fact knowledge to actions that were taken in good faith in the heat of the moment. Assume that the actions people were taking were based on the best available information they had at the time. Try to focus on how to provide more accurate information faster so everyone is better informed during your next incident response event.

**▶▶▶ POTENTIAL NEXT STEP**

**Make scripts (PowerShell or leverage some automation tooling) to do these actions in a rapid and consistent way.**

# 5 Implement Application Control

## SPECIAL NOTE

For most organizations, this section will likely be the trickiest to implement. Despite the difficulty, It will remain on this list because it is exceptionally effective at stopping many attacks.

## WHY YOU NEED IT

Application control (used to be called application whitelisting) is a technology that allows you to restrict the applications that are allowed to run on a machine. This prevents attackers from running malware they place on your system.

Important note: Like all security controls, skillful attackers do know how to get around application control. However, it is not easy. Not every attacker will be able to bypass it. Most often, in doing so, they generate a lot of noise. Once you have application control enabled, you can start looking for odd behaviors. One of the best detections is to alert when users are looking for application control tools running.

## HOW TO DO IT

If you have a third-party tool that can handle application control, please consider using that. If you do not, Windows has a built-in application control tool called AppLocker. All versions of Windows 10 and 11 now support this powerful feature. Microsoft has a guide for how to implement AppLocker. Carefully follow these instructions in a non-production environment before deploying to production systems.

## TIPS TO AVOID COMMON MISTAKES

### Profile on Actual Use

As powerful as these tools are, if you make a mistake in the configuration (block an application that is needed) you can prevent the system from working as needed by the user. The fastest and safest way to make a configuration is to base it on actual user behaviors.

If you have an application which tracks user behaviors, please leverage that. Some common third-party tools that do this are Endpoint Detection and Response (EDR) agents, Managed Detection and Response (MDR), as well as forensics agents.

If you do not have any of these products, another option is to take advantage of a little known feature in Windows. All applications used are tracked through the Windows System Resource Usage Monitor (SRUM). It contains a rolling 30-day list of the applications that were run by all users and the system. By using specialized tools, you can read the SRUM database. Perhaps the easiest tool to use for reading the SRUM database is SRUM-Dump.

### Don’t Rush to Enforcement

Once you’ve built the application control policy, it may be a good idea to run in “audit” mode or “learn” mode for a week or more. In this mode of operation, you will learn of the applications that would have been blocked.

# 6 Make This a Sustainable “Steady State”

## WHY YOU NEED IT

We’re going to show you how to leverage the above techniques to create a sustainable state that allows you to have success built upon success.

## HOW TO DO IT

Implementing robust security controls is a marathon, not a sprint. Pace yourself. Many organizations will attempt to suddenly “get serious about security” and try to fix everything all at once. Not only is this disruptive to ongoing issues, it rarely works well. Just like any personal fitness program, you tend to get better results by doing small impactful changes that result in incremental progress over time.

## TIPS TO AVOID COMMON MISTAKES

### Removing Noise Allows Focus

We’ve specifically focused on the defensive and detective controls that, in addition to being effective, will REMOVE NOISE from your environment. This, in turn, will allow you to focus more on the things that matter (specifically, removing even more noise!).

### Protect Your Process Improvement Time

Pretty soon, you will have created a positive feedback loop where each time you create better visibility and controls, you will be rewarded with more time to improve things. If you’re in a situation where you feel that “everything is on fire” the techniques we’ve covered in this document are for YOU. It may sound strange, but forcing a 2-hour block where you can focus and make tangible improvements on one of these controls will actually pay back multiple times.

### ▶▶▶ POTENTIAL NEXT STEP

**This paper is a resource allowing you an onramp to an effective security program. Consider going back through this list multiple times addressing the elements you needed to skip in the interest of time. Once you have comfort with this list, move to a deeper and more complex framework like the Center For Internet Security Critical Controls.**

## About the Authors

### Mick Douglas, SANS Principal Instructor

Mick Douglas has over 10 years of experience in information security and is currently the Managing Partner for InfoSec Innovations. He specializes in PowerShell, Unix, Data Visualization, Hardware, and Radio Hacking and teaches SEC504: Hacker Tools, Techniques, and Incident Handling and SEC555: SIEM with Tactical Analytics. He has also been on the GIAC Advisory Board for over 12 years. He's authored numerous tools, led pen-testing teams, and has consulted with some of the largest InfoSec companies. READ MORE ▶

### Jon Gorenflo, SANS Certified Instructor

When Jon Gorenflo took a job as a network administrator early in his career, the security responsibility of the position was mentioned almost as an afterthought. However, " it didn't take long for me to realize I was spending 90 percent of my time administering security products," he says. With no seasoned security veterans internally for Jon to lean on, he collaborated with teammates and found mentors by taking SANS classes, reading the SANS Pen Test blog and listening to the Security Weekly Podcast. READ MORE ▶

# Appendix

## PRE-AUTHORIZATION TO TAKE SYSTEM OR NETWORK ZONE OFFLINE

### System authorization

To prevent an entire organization attack, the [TEAM NAME] is permitted to take the [SYSTEM NAME] offline if they in their judgment believe it poses a harm to the rest of the organization.

[Date]
[System owner, title]
[Response team person, title]

### Network authorization

To prevent an entire organization attack, the [TEAM NAME] is permitted to take the [NETWORK ZONE NAME] offline if they in their judgment believe it poses a harm to the rest of the organization.

[IMPACTED system owners, title]
[Response team person, title]

## REASON SYSTEM/NETWORK WAS TAKEN OFFLINE

I, [Response team person, title] took the [system/network name] offline at [DATE AND TIME] because I believed it posed a risk to the rest of the network. I notified [system owner] at this time.

I based my decision on the following information.

[Justification here]

As part of process improvement, I will review this with [System owner, any other people] at [Future date, not more than 1 week out].

[Response team person, title][DATE]