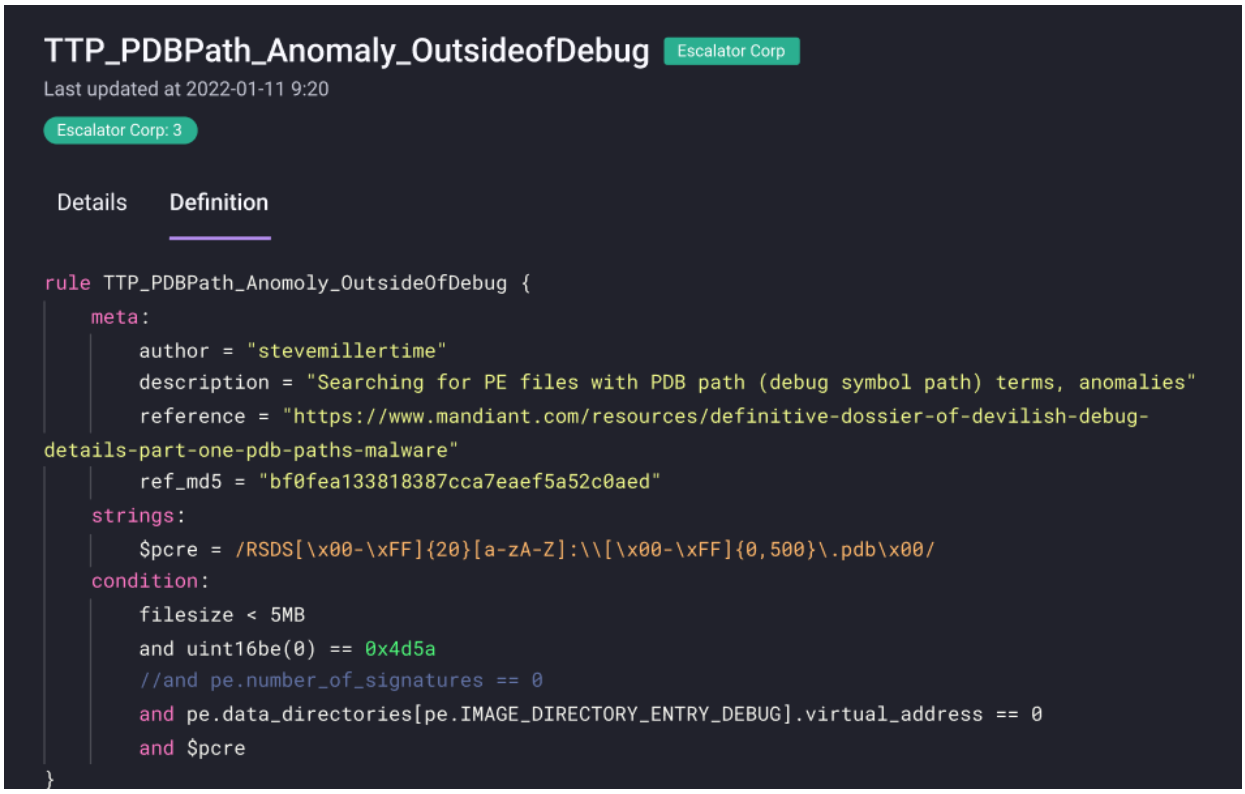
Blog • January 11, 2022

What is YARA and why it matters



Stairwell

If you're reading this, you're probably already bamboozled with security tooling. You've got all the products and you've heard all the pitches, yet you can't help but feel like something is missing in your life. You long for more flexible technology. You yearn for faster, more functional analyst tooling. If you're lucky, the feeling of emptiness, the void in your life, and the gaps in your security apparatus could be filled with YARA.

Yet Another Ridiculous Acronym (YARA) is indeed a ridiculous acronym in and of itself, but YARA is also a wonderful and immensely powerful open source security technology that helps analysts do pattern matching on files. What Snort does for packets and Sigma does for logs, YARA (more or less) does for files.

Pioneered and still developed by @plusvic of @virustotal, YARA helps researchers identify and classify malware samples. In its most basic implementation, an analyst composes a rule in YARA's own expression language to describe patterns and Boolean logic for matching conditions (an example is shown below). YARA rules are accordingly smashed against a set of files in YARA's internal engine to determine if any of the files match any of the rules, and then rule match details are provided as results. This seemingly simple functionality lays the groundwork for complex and elaborate implementations in a multitude of grander security systems and technologies.

```
rule silent_banker : banker
{
    meta:
        description = "This is just an example"
        threat_level = 3
        in_the_wild = true

    strings:
        $a = {6A 40 68 00 30 00 00 6A 14 8D 91}
        $b = {8D 4D B0 2B C1 83 C0 27 99 6A 4E 59 F7 F9}
        $c = "UVODFRYSIHLNWPEJXQZAKCBGMT"

    condition:
        $a or $b or $c
}
```

Like a fine Cabernet, YARA has gotten better with age. YARA has ebbed and flowed over the years into something broader and more flexible for more security purposes, and dozens of major security vendors have taken part through adoption and contributions to the open-source technology. YARA is almost everywhere you look, but you wouldn't know it. Behind the scenes, it is silently powering features of antivirus, next-gen firewall, endpoint detection and response, network detection and response, malware sandbox, and threat intelligence products and platforms around the world. In our opinion, YARA is one of the best-kept secrets in the security industry. While many vendors have capitalized on its potential, we think organizations can have more control over their destiny by peeling back the opaque covers to their security technology, and analysts can have more impact (and more fun) by using YARA more directly.

YARA is one of those technologies that is as powerful as you make it. If you put garbage into it, all you will get is garbage out of it. Yet if you feed it good data and fill it with bright ideas, the results can be magnificent and game-changing for your security work.

YARA for detection

Most people familiar with YARA and detection will think of YARA rules as "signatures" meant to characterize the attributes of known bad files or malware. These detection rules often contain strings and hex sequences and features of malware codified into rules, allowing automated systems to identify when malware is seen and classify it. There are high-fidelity YARA rules for malware families such as SUNBURST and CobaltStrike Beacon, and when precise rules like these match on files, the rule matches serve to alert security systems and people that something bad has shown up.

Security vendors often use precise YARA rules to generate detection events and alerts, and behind the green curtain of their products, they use broader, less-specific rules to collect and group subsets of data, take measurements, apply labels, and hunt through the data in ways that many users cannot see. One reason for the rise in open source detection technologies is that analysts wish to openly share and exchange rules to automate detection research in formats that transcend particular products. They wish for their ideas and research to have an automation vehicle. This is where YARA rules can shine, and there is a growing community of analysts who share and exchange detection rules in the public sphere to help spread detection ideas in formats that are structured

Signatures often get a bad rap for being brittle, overly simplistic, easy to bypass, or prone to false positives or false negatives — and sometimes they are. But content matching and conditional Boolean logic in any rule format serve as the basis for most detection technologies. Signatures aren't bad things. They're just logical rules and can be as complex and creative as their authors. And, within a detection technology, the most complex and creative rules can still only be as elegant and effective as their technological implementations.

YARA for incident response

Beyond searching for known-bad patterns, YARA can be used to speed up an investigation. If you find a sample of malware on a computer in your purview, you can create a new YARA rule and run it against files from your enterprise assets such as your endpoints, which may help you better scope the incident and figure out where you need to dive deeper. If you're combatting a threat actor that loves to use encrypted RAR files, you can crank out a rule for that and run that against new files created on web servers or against email attachments to find places where data may be staged for exfiltration.

In more creative implementations, incident responders sometimes use EDR tools to collect volatile memory from endpoints and run specialized YARA rules against the memory dumps to find malware that is elusive or simply no longer present on disk. Similarly, you could do this for network data, where you apply special YARA rules against packet payloads or whole network streams re-assembled into a file.

YARA for labeling and context

YARA is much more than a typical detection technology because YARA rules do not always have to describe something that is malicious. Analysts can use YARA to describe almost anything that may be present in file data, and technologies around the world leverage descriptive rules to apply labels and offer different types of context to data of all sorts.

You can write a rule to identify particular file formats, unsigned drivers, documents with specific authors, or executables with anomalies of all sorts. In essence, when you write a

Because rule matches serve as labels, when YARA rules are run against files, they can serve to accelerate analysis by providing crucial context to security teams. For example, if you run YARA rules against email attachments, rule matches may serve as a label that a given file is an “encrypted zip” or another “document with VBA macro.” These pieces of context help analysts make quick decisions on files and may improve the quality of analysis or investigation.

Labeled data becomes more important over time as the labels become weak signals that can be combined together or with other file features for a more profound impact. Imagine searching for unsigned drivers that have a low global prevalence and were first seen in the last week. This idea may start out with a simple YARA rule as the foundation, but then incorporate the other measurements for a tighter, more elegant result.

Summary

YARA lives in the space where intelligence meets automation. If you’re a manager or a CISO, you can think of “logical rules” in YARA as *automations*. YARA rules are effectively a form of human analysis that is codified, re-written in special syntax as code. These rules are one way to capture and store human analytical conclusions and knowledge about threat actors, malware, or attributes of complex technical data.

When analysts invent ways to identify or detect a particular thing, they store that knowledge in a format that helps them operationalize across data at rest (in a data store), or on data in-motion (passing across a sensor). This makes YARA a powerful technology because it stores intelligence and serves as an automation vehicle. When analysts leave your team or organization, their work is stored indefinitely and can be applied at scale to data in motion or at rest as you see fit.

For analysts who spend time studying threat actors, malware, and intrusions, it can be frustrating to see your work distributed in short-lived forms, and disheartening to watch your research grow old and die in tickets and wikis that future generations may never read. Because YARA is beloved and used by many organizations, and because so many use it or rely on it behind the scenes, anything you put into YARA rules can be quickly and easily shared and operationalized on tech and data of all sorts. By applying your research and insights into things like YARA rules, your hard work may have a longer-

Remember that YARA is just a tool, and like all tools, it has its strengths and limitations. Yet across the practices of detection, incident response, intelligence analysis, file classification, and more, YARA is a fast and easy automation vehicle for pattern matching on file content. Like any flexible technology, you can use the expressive rule syntax to write the most basic of signatures, pen an Odyssean epic, or anything in between. The power and creativity of the technology rely on the power and creativity of the analysts that use it, but what's more is that when YARA is leveraged effectively, the work put into it will be easily operationalized and long lasting. We believe investing time into YARA will pay off in the long run, and we hope that this post inspires you to think about flexing on YARA in new and different ways.

If your interest is piqued, and you'd like to go deeper technically, please see Quick n' dirty detection research: Building a labeled malware corpus for YARA testing.

Steve Miller

SR. RESEARCHER

YARA Warlock

Steve is a researcher focused on adversary tradecraft, the TTPs or modus operandi of threat actors. He loves malware, pcap, detection, and collecting modular synthesizers in his beat laboratory.

Stay connected

Privacy Policy / Terms / Security

Cookie Preferences