

THE (LIFE AND) DEATH OF THE PATTERN FILE

David M. Perry
Trend Micro, USA

Tel +1 714 846 5689
Email david_perry@trendmicro.com

ABSTRACT

Recent reports list 2007 as the year of the greatest number of tracked threats in history. Vendors report increases of up to 5,000 new actual malware samples per day. Some botnets are replaced in the wild every couple of minutes. Does this mean that the pattern file will soon go the way of the passenger pigeon? What alternative technologies are announced by the industry to supplant the pattern file? Will this general trend continue into the future?

INTRODUCTION: RECENT CRITICISM OF PATTERN FILE USE

A recent article makes a point of claiming that the anti-virus industry and anti-virus products are no longer an acceptable whole solution. The fact that those of us inside this industry haven't made pure AV scanners in more than a decade is apparently beside the point. The author then goes on to talk about the specific practice of identifying viruses by hand, and downloading those definitions is the particular archaic practice that is being proscribed.

I do not need to quote the exact article, because it is only one of many that go back to the very beginning of the industry. We will talk about and quote some of this criticism in the course of this paper, but the criticism can be reduced to several main points:

1. The AV industry is dead/commoditized/outdated.
2. The AV scanner is not sufficient for today's threat landscape.
3. The pattern file method is obsolete and should be replaced by something shinier/better/more advanced (choose buzzword).

The description of the technology assumed to rescue us from the problems with pattern files will change every three years to something new.

THE REALITY: IT'S NOT ALL THAT SIMPLE

Now, truth be told, it is actually possible that this time the naysayers may be right, we may actually be seeing the approach of the end of the pattern file as the lead technology in the fight against malware. Basic pattern recognition has been the mainstay and staple of malware detection for 20 years. It is flexible to many situations, and perhaps most importantly, it is the most cost-effective way for a manufacturer to provide detection. It is all too easy to conflate pattern recognition from the mainstay to the *only* thing, and we seem to want to see most things as a sole

and simple idea. But the past, present and future of virus detection is just not that simple.

There are generic patterns, inoculation, heuristic detection, behaviour blocking, quarantine, sandboxing and other methodologies too numerous to mention. Modern anti-malware must accomplish emulation, decryption and host-based intrusion detection, must couple with a firewall, must help detect fraud and spam, must use reputation, white and blacklisting, and many other things as well. Today, anti-malware still uses pattern files, but are their days numbered? Is this the end of the pattern file? Only time will tell.

THE PATTERN FILE, A HISTORY

From the very beginning, AV software has used a variety of methods to detect, remove, quarantine, block and delete malware. Some of it was appropriate to the malware involved, other parts of it sensitive to the fashions in OS and application, and other parts just peculiar to one vendor or another.

The pattern file itself began very simply. At first it was only occupied with detection (some programs, such as *McAfee SCAN*, sought only to detect) but eventually repair scripts, virus names and other data were incorporated with or linked to the pattern file, creating an ever more complicated array of patterns.

Many early scanners used specific strings in hex to do the detection, and some even offered the option of a user entering his own strings, thus creating his own pattern files. This practice continues today in the open source *Clam AV* project.

Other scanners used algorithms that were matched with compiled or modified strings that allowed faster pattern matching, more certain matching or other proprietary engine support. This describes the majority of today's pattern file and engine combinations.

As the number and types of malware grew more numerous and complex, both pattern files and engines needed to handle more and more different types and styles of detection. This has changed many of the earliest assumptions made by pattern scanning utilities. Chief among these was the idea of infectable file types.

Pattern files are also known as signature files, definition files, string files, update files and so on. Every anti-malware manufacturer still uses them, to a greater or lesser degree.

WHAT OTHER METHODS HAVE BEEN USED BESIDES PATTERN FILES?

As originally defined, viruses needed to attach themselves to some other executable file. This was usually either the boot sector of some disk or a program file. In this environment, it was popular to use a technique known as inoculation to tell if a file had been changed in size or if it had been tampered with. Each manufacturer had its own method of inoculating files, sometimes appending a wrapper to the file, sometimes making a covert database containing measurements of the known files. This process was not entirely separate from the pattern files of the day – sometimes a vendor would combine information from the inoculation and the pattern file, and sometimes the inoculation

data was crucial to applying a repair or clean script called out in the pattern file.

Generic pattern files might look for a specific item thought of as highly suspicious when taken in context. The perfect example of this was the *McAfee* gen/b detection from the mid-1990s. Any boot sector containing the command set to write to a disk, or any calls to Disk I/O inside the MBR were seen as generic evidence of a boot sector virus (MBR virus).

Heuristics might use rules of thumb to determine a specific kind of virus or other malware.

Contemporary ‘bags of tricks’ might include whitelisting, HIPS, web reputation, IP reputation, file reputation and other methods.

Every bit of the history of malware detection is rich with unique and seasonal methods, and today more so than ever. This reflects the massive growth of malware as an idea, as a phenomenon, and finally as a criminal industry.

HOW HAS THE VIEW ON PATTERN FILES CHANGED?

To many of the earliest end-users, the very existence of an update was a stark surprise. Every day in that first decade at least one call would come in from a user who was astonished to learn that the product needed to add new data to catch the latest viruses. This caused a number of users to believe that there was something dodgy about the whole business.

There have been periodic attempts to release ‘no update’ anti-malware detection programs. There were, in the past, three main problems with these programs. First, their ‘patternless detection’ was rigidly married to the malware schema present at the time, with neither good historical detection nor good ability to move to the next malware fad. The second fault was a tendency to create false positive detections. Finally, an all-heuristic scanner is incapable of naming the exact malware found, and this was, for a given time in the industry, a fatal flaw.

By the time the average user had accepted the reality of constant updates, the industry had acquired a number of fierce critics. Perhaps the most colourful of these was Rob Rosenberger. On his website, Rosenberger criticized AV companies for keeping their customers ‘addicted’ to updates. Rob clearly loved his analogy of addiction, and repeated it over and over. He was certain that every company in the AV industry had a secret *X Files* project that would universally detect without the use of any updates. It was pointed out to Rob that updates cost vast amounts of money to produce and brought no additional revenue to the industry.

PATTERN FILE DELIVERY SYSTEMS

The earliest pattern files I recall were at *Norton*. The patterns were actually entered by hand into the product, in alphanumeric base 36 code. These patterns were not actual strings, and there was no allowance for users to ‘make up their own patterns’ as there was at the time at *McAfee*. Delivery of these patterns was arcane. One could either elect to have the file faxed to the user, or have it read to them over the telephone by a member of tech support. Either way, the user ended up typing the update by

hand into his individual copy, then adding a checksum to ensure accuracy.

Over the hill at *McAfee*, the earliest patterns were delivered in a recompiled scanner. The entire package was recompiled with new data (definition) files, and then placed on the ‘locked’ *McAfee* bulletin board. In the very early days, a chain of ‘Associates’ also allowed the updates for download from their bulletin boards. This was referred to as ‘Electronic Software Distribution’ (ESD) and was universally seen as a real innovation.

Trend Micro had a system of pattern file tracking that was good up to the first thousand pattern files; it was amazing how the effort to upgrade all users smoothly illuminated the starkly accelerating pace of needed updates.

For years, users could subscribe to diskette-based updates, which arrived first quarterly, then monthly, then weekly. Disk-based updates seemed like a good revenue generator for the companies involved but the logistics eventually overwhelmed the promise of subscription fees.

The rise of the online services *CompuServe*, *AOL* and *Prodigy* also made for great download bases, as well as a good forum for customer service and technical support.

Finally, the Internet became the preferred delivery method not only for pattern files, but for most of the data in the world. Original web page downloads gave way to client proprietary updates, and finally to the ‘no-brainer’ updating system we enjoy today.

There isn’t a product, or a company, or an individual in this entire industry that hasn’t had to go through rapid deployment, vast technological change and customer confusion in the matter of updates, pattern files or whatever they are called in your neck of the woods. This is one of the core issues that we all deal with.

By now, it should be obvious to all that with our research labs bursting at the seams, and the constant chatter of update packets filling every net junction in the world, the security industry is currently and largely a delivery service for all manner of update files. Pattern files are as common on the Internet as MP3 files, as pervasive as spam. This is the first clue as to what might become the next change in our world.

CHANGES IN THE THREAT LANDSCAPE

It is said too often. The number and type of malware agents has expanded beyond all imagined capacity. The startling revelations of how many new items are appearing per day, how many total unique patterns there are, etc. are being delivered so frequently that we have become totally numb to any sense of perspective such revelations are meant to impart. There are a helluva lot of trojans, keyloggers, bots, viruses, worms, trojans, poisoned DNS servers, bad websites, spam – there are a lot of articles books and experts.

Much of the self-replicating code has become self-modifying to the point that a simple string can no longer define it. Most of the non-replicating malcode is taken out of circulation in a matter of minutes (soon to be seconds). These facts are evidence of countermeasures taken against us as an industry. It goes without

saying that the bad guys on the other side of this equation are an industry of their own.

There are way too many pattern files, and they are not deployed quickly enough to make a difference in prevention.

PATTERN FILES: YOUR DAYS ARE NUMBERED

Amazingly, the biggest problem with updating the client has nothing to do with the efficacy in detection, prevention and remediation of malware. The biggest problem is that with the current exponential (or, if you prefer, geometric) growth of malware to defend against, the sheer size and frequency of updating will become a worse problem than the malware itself.

That's right, the pattern file is so big and so frequent that, especially in the enterprise-level installation, it generates too much network traffic.

WHAT THINGS MAY COME

This is not a problem for any vendor in particular; this is an industry-wide problem. Some vendors will move detection into the cloud. Some will develop distributed processing. Some applications will pick up a part of the burden. But in that brave new world, there will be one thing you can count on finding in almost every anti-malware or content security niche: pattern files!

PATTERN FILES?

Some items will always need to be positively identified. Some detections will need to refer back to an Internet-based remediation service. Some things will need to be updated to match the ever-changing threat. Updates will appear in a reduced place of prominence, but never in a place of reduced importance, so in conclusion I have only this to say:

The pattern file is dead! Long live the pattern file!