



# SE Labs

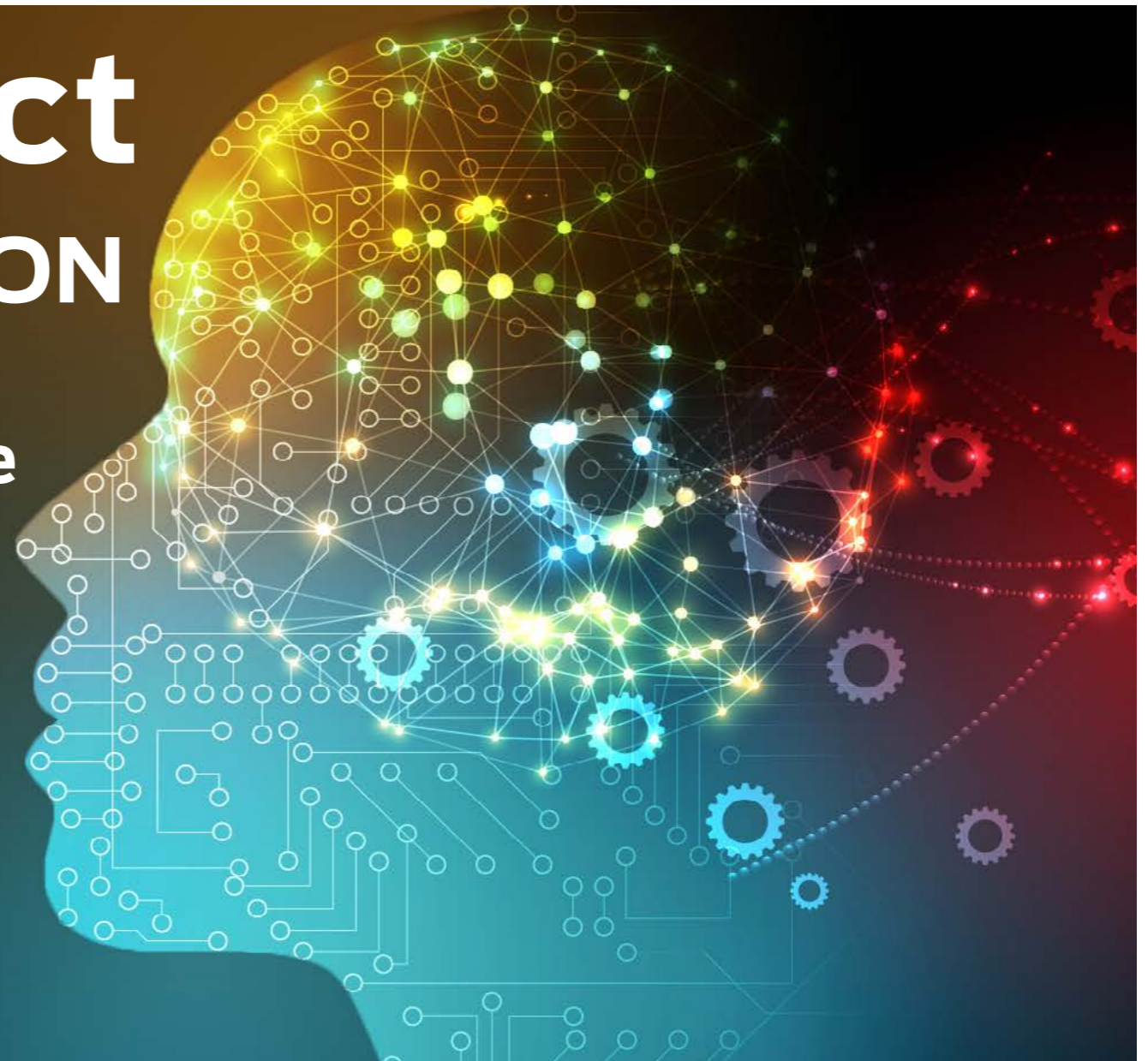
INTELLIGENCE-LED TESTING

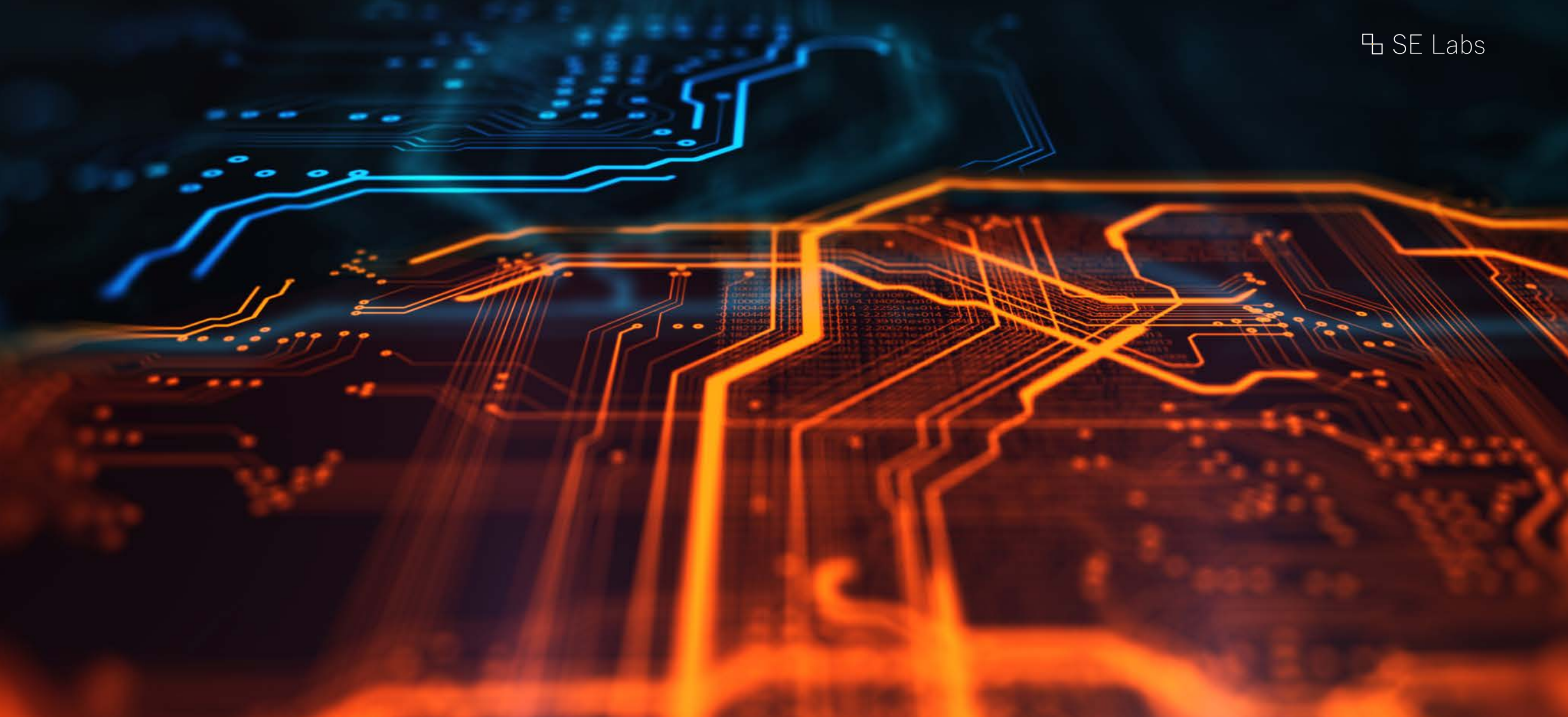
## Deep Instinct

### THREAT PREVENTION EVALUATION:

Sophisticated, high-profile  
file-based and file-less  
targeted attacks

February 2019





SE Labs tested **Deep Instinct D-Client** against a range of high-profile, known malware campaigns and a selection of new, sophisticated and unknown targeted attacks.

The tested version of **D-Client** contained a deep learning-based system that was trained six months prior to testing.

**MANAGEMENT**

**Director** Simon Edwards

**Operations Director** Marc Briggs

**Office Manager** Magdalena Jurenko

**Technical Director** Stefan Dumitrascu

**TESTING TEAM**

Thomas Bean

Dimitar Dobrev

Liam Fisher

Gia Gorbald

Pooja Jain

Ivan Merazchiev

Jon Thompson

Dave Togneri

Jake Warren

Stephen Withey

**IT SUPPORT**

Danny King-Smith

Chris Short

**PUBLICATION**

Steve Haines

Colin Mackleworth

**Website** [www.SELabs.uk](http://www.SELabs.uk)

**Twitter** @SELabsUK

**Email** [info@SELabs.uk](mailto:info@SELabs.uk)

**Facebook** [www.facebook.com/selabsuk](http://www.facebook.com/selabsuk)

**Blog** [blog.selabs.uk](http://blog.selabs.uk)

**Phone** 0203 875 5000

**Post** ONE Croydon, London, CR0 0XT

SE Labs is BS EN ISO 9001 : 2015 certified for  
The Provision of IT Security Product Testing.

SE Labs is a member of the Microsoft Virus Information  
Alliance (VIA); the Anti-Malware Testing Standards  
Organization (AMTSO); and the Messaging, Malware  
and Mobile Anti-Abuse Working Group (M3AAWG).

# CONTENTS

|  |    |
|--|----|
| Introduction                             | 04 |
| Executive Summary                        | 05 |
| 1. Public Threats by Family              | 06 |
| 2. Public Threats by Individual Campaign | 07 |
| 3. Script-Based Targeted Attacks         | 08 |
| 4. Microsoft Office Format-Based Attacks | 09 |
| 5. Shellcode Injection Attacks           | 10 |
| 6. Legitimate Software Handling          | 10 |
| 7. Conclusions                           | 11 |
| Appendix A: Terms Used                   | 12 |
| Appendix B: FAQs                         | 12 |
| Appendix C: Product Versions             | 12 |

Document version 1.0 Written 25th February 2019



## INTRODUCTION

# Enemy Unknown: Handling Customised Targeted Attacks

## Detecting and preventing threats in real-time

Computer security products are designed to detect and protect against threats such as computer viruses, other malware and the actions of hackers. A common approach is to identify existing threats and to create patterns of recognition, in much the same way as the pharmaceutical industry creates vaccinations against known biological viruses or police issue wanted notices with photographs of known offenders.

The downside to this approach is that the virus or criminal has to be known to be harmful, most likely after someone has become sick or a crime has already been committed. It would be better to detect new infections and crimes in real-time and to stop them in action before any damage is caused. This approach is becoming increasingly popular in the cyber security world.

**Deep Instinct** claims that its **D-Client** software is capable of detecting not only known threats but those that have not yet hit computer systems in the real world. Determining the

accuracy of these claims requires a realistic test that pits the product against known threats and those typically crafted by attackers who work in a more targeted way, identifying specific potential victims and moving against them with speed and accuracy.

This test report used a range of sophisticated, high-profile threat campaigns such as those believed to have been directed against the US Presidential election in 2016, in addition to directing more targeted attacks against the victim systems using techniques seen in well-known security breaches in recent months and years.

The results show that **Deep Instinct D-Client** provided a wide range of detection and threat blocking capability against well-known and customised targeted attacks, without interfering with regular use of the systems upon which it was deployed. The deep learning system was trained in August 2018, six months before the customised targeted threats were created.

# Executive Summary

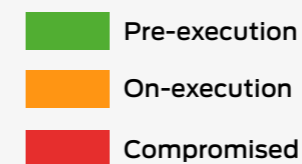
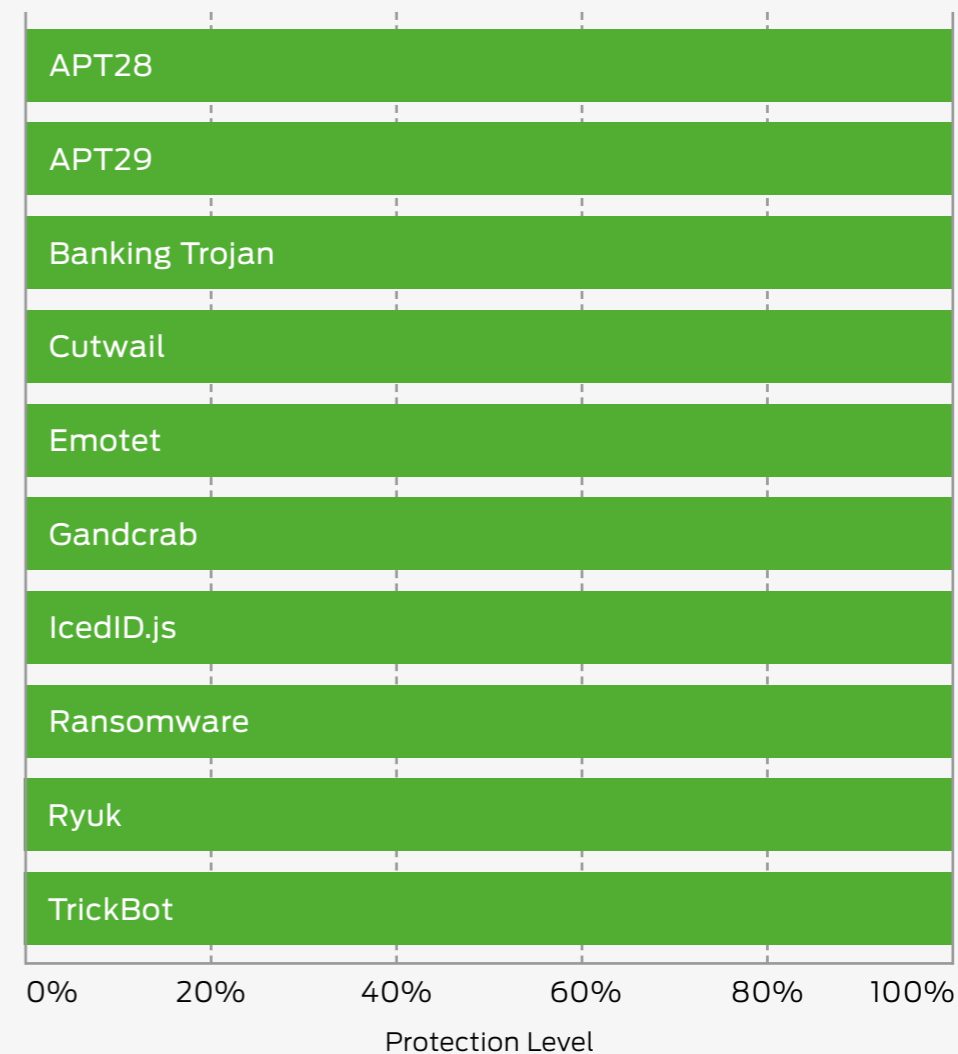
Deep Instinct D-Client, and endpoint protection product, was exposed to a range of attacks, including:

- Malware from well-publicised, impactful breaches
- Script-based (aka 'file-less') targeted attacks (e.g. JavaScript files)
- Attacks using exploits targeted at Microsoft file format vulnerabilities (e.g. malicious Microsoft Word documents)
- Targeted shellcode injection attacks

Legitimate files were used alongside these malicious files to measure any false positive detections or other sub-optimum interactions.

**D-Client** detected all of the variants of the public attacks and each of the targeted attack components. It also protected the targets from these same attacks, preventing them from providing remote access, causing damage or stealing data.

## Threat Prevention Details

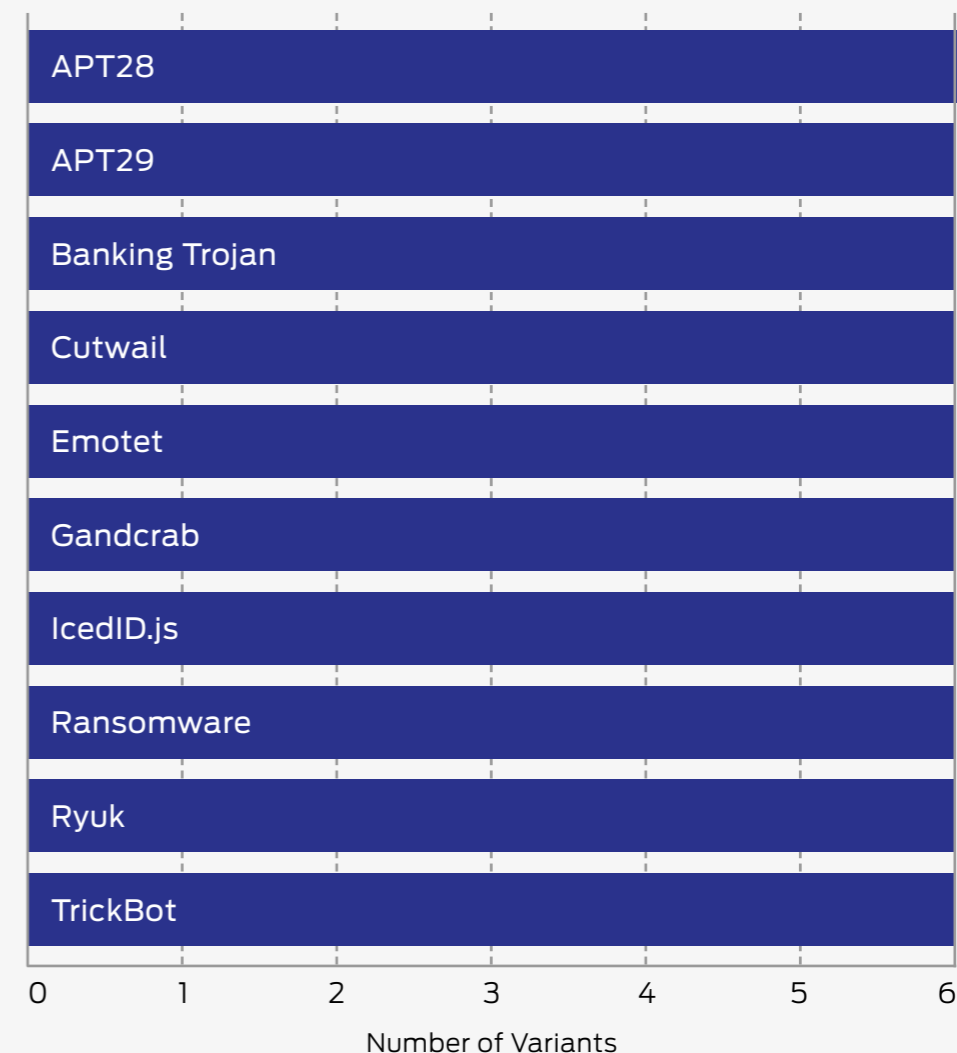


# 1. Public Threats by Family

These threats were discovered before the test was run and have been analysed and otherwise researched for between three to 36 months. They represent high-profile breaches widely reported on over the last three years.

| PUBLIC THREATS |  |
|----------------|--|
| Campaign       | Details  |
| APT28          | Believed to be run by Russian intelligence, APT28 (aka Fancy Bear) was reportedly behind an attempt to interfere with the US Presidential election in 2016.<br>Ref: <a href="https://attack.mitre.org/groups/G0007/">https://attack.mitre.org/groups/G0007/</a>  |
| APT29          | Attributed to the Russian Government, and otherwise known as Cozy Bear, APT29 has operated since at least 2008 and is thought to be behind the compromise of the Democratic National Committee in 2015.<br>Ref: <a href="https://attack.mitre.org/groups/G0016/">https://attack.mitre.org/groups/G0016/</a>                            |
| Banking Trojan | This series of banking Trojan attack involved achieving persistence on the target and moving across the network uses the Remote Desktop Protocol (RDP).  |
| Cutwail        | The Cutwail botnet, a network of infected systems designed to give attackers control of the resources belonging to unwitting victims, was at one time believed to be the largest on the internet. It is frequently used to send email spam.  |
| Emotet         | This advanced bank Trojan is often used to distribute other banking Trojans. It copies itself over networks and is considered by the US Department of Homeland Security to be, "among the most costly and destructive malware affecting state, local, tribal, and territorial (SLTT) governments, and the private and public sectors." |
| Gandcrab       | This ransomware campaign was new at the time of testing. Frequent and fast changes to the code suggest that its developers are putting a lot of effort into maintaining the software and attempting to evade detection. It was the first ransomware to request payments in the DASH crypto currency.                                   |
| IcedID.js      | IcedID.js is a banking Trojan that targets banks, payment card providers and e-commerce sites, among other victims. It targets business users, whose log-in details are more valuable than those of average internet users.  |
| Ransomware     | Variously known as Aurora and Zorro ransomware, this group of attacks took place at the time of testing. It is thought to encrypt the disks of victims unless they are located in Russia.  |
| Ryuk           | Unlike many ransomware attacks, Ryuk was targeted at specific victims. At the time of writing these were the desktop computers, servers and data centre systems belonging to large enterprises. The ransom demanded was subsequently much higher than that usually received by individual victims.                                     |
| TrickBot       | Aimed at businesses worldwide, the TrickBot banking Trojan is designed to access internet accounts and steal personal information with the ultimate goal of committing fraud.  |

## Public Threat Campaigns Detection



## 2. Public Threats by Individual Campaign

Attackers develop their tools and malware to evade detection over a period of time. These tables show the breadth of the tested product's detection and protection capabilities when facing a range of threat variants.

| CAMPAIGN: APT28 |          |         |
|-----------------|----------|---------|
| Threat Variant  | Detected | Blocked |
| APT28 1         | ✓        | ✓       |
| APT28 2         | ✓        | ✓       |
| APT28 3         | ✓        | ✓       |
| APT28 4         | ✓        | ✓       |
| APT28 5         | ✓        | ✓       |
| APT28 6         | ✓        | ✓       |

| CAMPAIGN: APT29 |          |         |
|-----------------|----------|---------|
| Threat Variant  | Detected | Blocked |
| APT29 1         | ✓        | ✓       |
| APT29 2         | ✓        | ✓       |
| APT29 3         | ✓        | ✓       |
| APT29 4         | ✓        | ✓       |
| APT29 5         | ✓        | ✓       |
| APT29 6         | ✓        | ✓       |

| CAMPAIGN: BANKING TROJAN |          |         |
|--------------------------|----------|---------|
| Threat Variant           | Detected | Blocked |
| Banking Trojan1          | ✓        | ✓       |
| Banking Trojan2          | ✓        | ✓       |
| Banking Trojan3          | ✓        | ✓       |
| Banking Trojan4          | ✓        | ✓       |
| Banking Trojan5          | ✓        | ✓       |
| Banking Trojan6          | ✓        | ✓       |

| CAMPAIGN: CUTWAIL |          |         |
|-------------------|----------|---------|
| Threat Variant    | Detected | Blocked |
| Cutwail1          | ✓        | ✓       |
| Cutwail2          | ✓        | ✓       |
| Cutwail3          | ✓        | ✓       |
| Cutwail4          | ✓        | ✓       |
| Cutwail5          | ✓        | ✓       |
| Cutwail6          | ✓        | ✓       |

| CAMPAIGN: EMOTET |          |         |
|------------------|----------|---------|
| Threat Variant   | Detected | Blocked |
| Emotet1          | ✓        | ✓       |
| Emotet2          | ✓        | ✓       |
| Emotet3          | ✓        | ✓       |
| Emotet4          | ✓        | ✓       |
| Emotet5          | ✓        | ✓       |
| Emotet6          | ✓        | ✓       |

| CAMPAIGN: GANDCRAB |          |         |
|--------------------|----------|---------|
| Threat Variant     | Detected | Blocked |
| Gandcrab1          | ✓        | ✓       |
| Gandcrab2          | ✓        | ✓       |
| Gandcrab3          | ✓        | ✓       |
| Gandcrab4          | ✓        | ✓       |
| Gandcrab5          | ✓        | ✓       |
| Gandcrab6          | ✓        | ✓       |

| CAMPAIGN: ICEDID.JS |          |         |
|---------------------|----------|---------|
| Threat Variant      | Detected | Blocked |
| IcedID1             | ✓        | ✓       |
| IcedID2             | ✓        | ✓       |
| IcedID3             | ✓        | ✓       |
| IcedID4             | ✓        | ✓       |
| IcedID5             | ✓        | ✓       |
| IcedID6             | ✓        | ✓       |

| CAMPAIGN: RANSOMWARE |          |         |
|----------------------|----------|---------|
| Threat Variant       | Detected | Blocked |
| Ransomware1          | ✓        | ✓       |
| Ransomware2          | ✓        | ✓       |
| Ransomware3          | ✓        | ✓       |
| Ransomware4          | ✓        | ✓       |
| Ransomware5          | ✓        | ✓       |
| Ransomware6          | ✓        | ✓       |

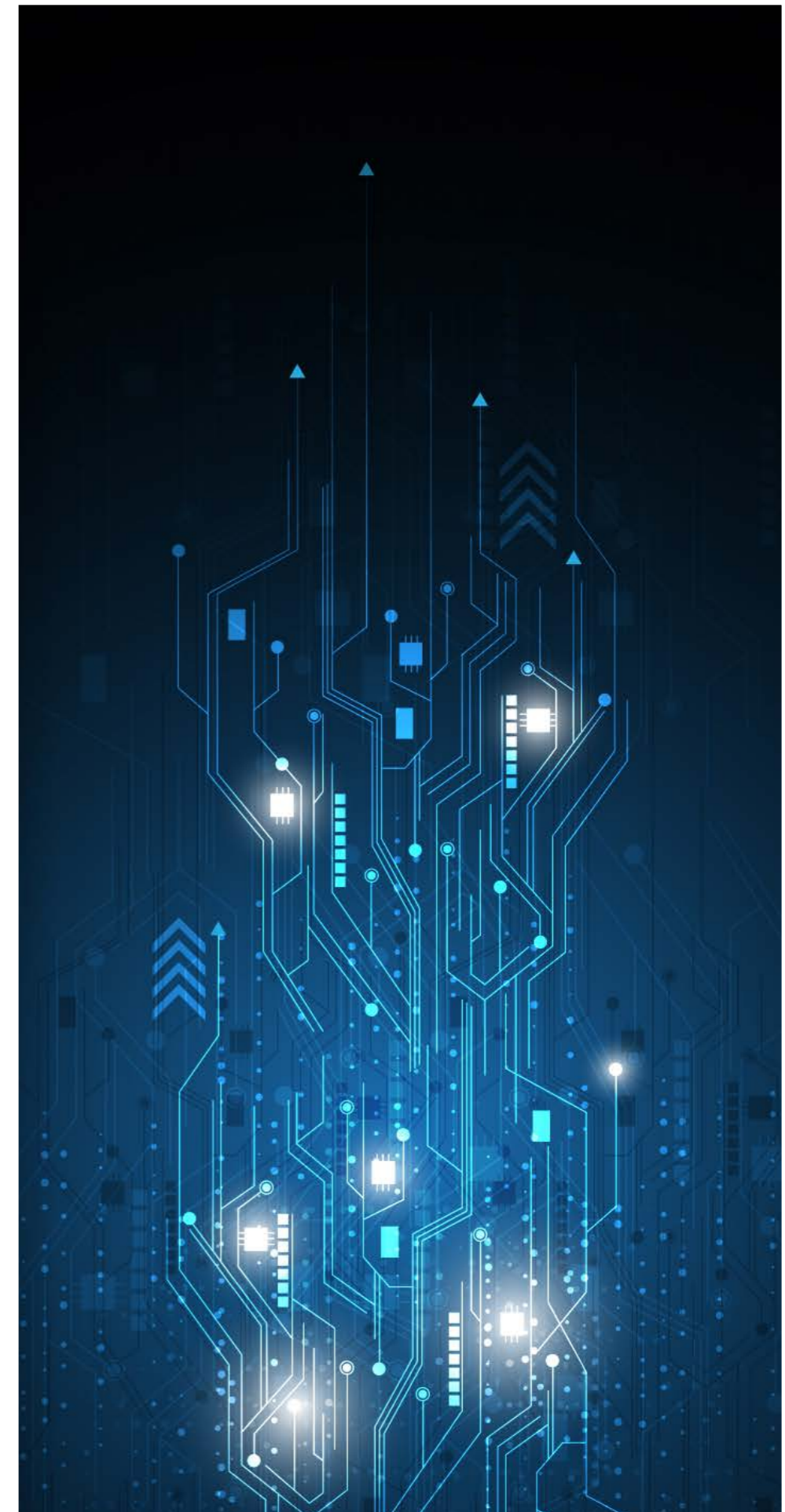
| CAMPAIGN: RYUK |          |         |
|----------------|----------|---------|
| Threat Variant | Detected | Blocked |
| Ryuk1          | ✓        | ✓       |
| Ryuk2          | ✓        | ✓       |
| Ryuk3          | ✓        | ✓       |
| Ryuk4          | ✓        | ✓       |
| Ryuk5          | ✓        | ✓       |
| Ryuk6          | ✓        | ✓       |

| CAMPAIGN: TRICKBOT |          |         |
|--------------------|----------|---------|
| Threat Variant     | Detected | Blocked |
| TrickBot1          | ✓        | ✓       |
| TrickBot2          | ✓        | ✓       |
| TrickBot3          | ✓        | ✓       |
| TrickBot4          | ✓        | ✓       |
| TrickBot5          | ✓        | ✓       |
| TrickBot6          | ✓        | ✓       |

### 3. Script-Based Targeted Attacks

So-called ‘file-less’ attacks rely less on standard malicious executables and involve injecting code directly into the target’s memory or embedding malicious code into scripts. This makes them potentially harder for some security products to detect and protect against.

| SCRIPT-BASED TARGETED ATTACKS                  |          |                  |         |
|--|----------|------------------|---------|
| Attack Type                                    | Detected | Full Remediation | Blocked |
| PowerShell Empire (Batch file launcher)        | ✓        | ✓                | ✓       |
| PowerShell Empire (Bash script launcher)       | ✓        | ✗                | ✓       |
| PowerShell Empire (One-line launcher)          | ✓        | ✗                | ✓       |
| PowerShell Empire (Visual Basic launcher)      | ✓        | ✗                | ✓       |
| PowerShell Empire (Batch file launcher; HTTPS) | ✓        | ✓                | ✓       |
| VBS (Encrypted HTTPS, heavy re-encoding)       | ✓        | ✗                | ✓       |
| VBS (Non-encrypted HTTP, heavy re-encoding)    | ✓        | ✗                | ✓       |
| JS (Non-encrypted TCP, heavy re-encoding)      | ✓        | ✗                | ✓       |
| JS (Encrypted HTTPS, re-encoding)              | ✓        | ✗                | ✓       |
| JS (Non-encrypted HTTP, re-encoding)           | ✓        | ✗                | ✓       |

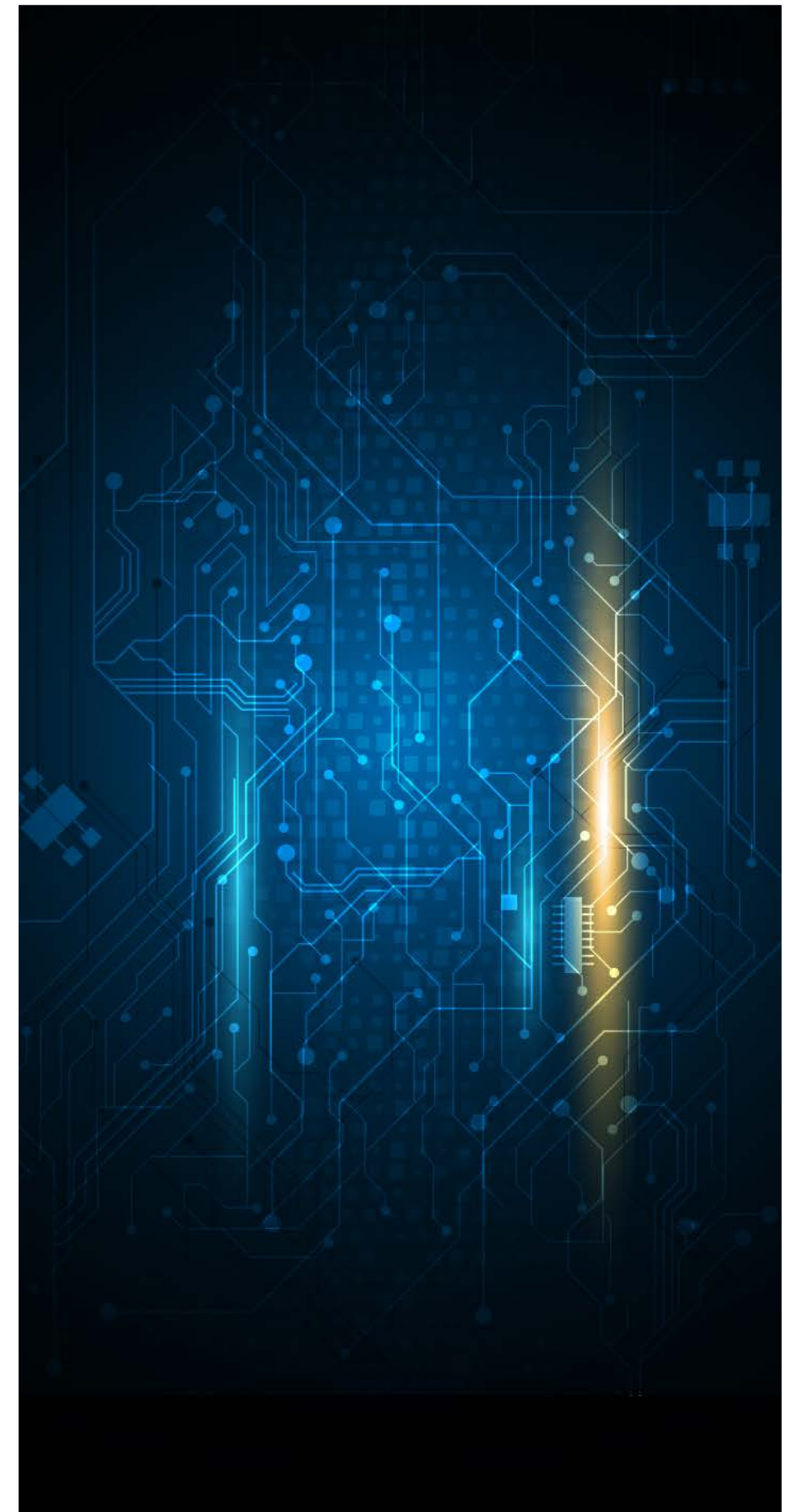




## 4. Microsoft Office Format-Based Attacks

These attacks exploit vulnerabilities in well-known Microsoft Office applications. The attacks appear to be regular documents but, when a target opens them, they execute malicious code and, in these cases, attempt to provide remote access to the attacker.

| MICROSOFT OFFICE FORMAT-BASED ATTACKS                     |          |         |
|---|----------|---------|
| Attack Type   | Detected | Blocked |
| Office Word Macro, Non-encrypted HTTP                     | ✓        | ✓       |
| Office Word Macro, Encrypted HTTPS                        | ✓        | ✓       |
| Office Word Macro, Non-encrypted TCP                      | ✓        | ✓       |
| Office Word Macro, Non-encrypted TCP (alternative method) | ✓        | ✓       |
| CVE-2017-11882, Non-encrypted HTTP                        | ✓        | ✓       |
| CVE-2017-11882, Encrypted HTTPS                           | ✓        | ✓       |
| CVE-2017-11882, Non-encrypted TCP                         | ✓        | ✓       |
| CVE-2017-0199, Non-encrypted HTTP                         | ✓        | ✓       |
| CVE-2017-0199, Encrypted HTTPS                            | ✓        | ✓       |
| CVE-2017-0199, Non-encrypted TCP                          | ✓        | ✓       |



## 5. Shellcode Injection Attacks

These attacks inject malicious code into legitimate processes on remote targets using different levels of encryption and injection methods, with a view to gaining remote access.

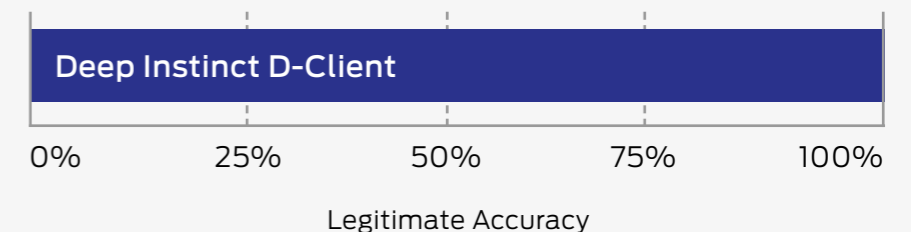
| SHELLCODE INJECTION ATTACKS  |          |         |
|--|----------|---------|
| Attack Type  | Detected | Blocked |
| Windows Shellcode Injection VirtualAlloc, Non-encrypted HTTP                 | ✓        | ✓       |
| Windows Shellcode Injection VirtualAlloc, Encrypted HTTPS                    | ✓        | ✓       |
| Windows Shellcode Injection HeapAlloc, Non-encrypted HTTP                    | ✓        | ✓       |
| Windows Shellcode Injection HeapAlloc, Encrypted HTTPS                       | ✓        | ✓       |
| Windows Shellcode Injection Process Inject, Non-encrypted HTTP               | ✓        | ✓       |
| Windows Shellcode Injection Process Inject, Encrypted HTTPS                  | ✓        | ✓       |
| Windows Shellcode Injection Process Inject, Non-encrypted HTTP (svchost.exe) | ✓        | ✓       |
| Windows Shellcode Injection Process Inject, Encrypted HTTPS (csrss.exe)      | ✓        | ✓       |
| Windows Shellcode Injection Thread Hijack, Non-encrypted HTTP                | ✓        | ✓       |
| Windows Shellcode Injection Thread Hijack, Encrypted HTTPS (svchost.exe)     | ✓        | ✓       |

## 6. Legitimate Software Handling

It is necessary, when testing a security product's ability to handle threats, to also measure how it handles legitimate code. Failure to do so means that a product that blocks both good and bad effectively will win a test but cause extreme disruption in the real world.

In this test we measured any incorrect classifications of files already present on the system, including many thousands of legitimate files from Microsoft and other third parties.

There were no 'false positives' and no other types of sub-optimum handling of legitimate files.



## 7. Conclusions

This test was designed to examine **Deep Instinct's** claim that its **D-Client** endpoint software was capable of detecting and blocking known and unknown cyber threats including file-based and file-less attacks.

To test that claim SE Labs collected malware from a range of well-publicised breaches, including attacks from the APT28 (Fancy Bear) group that reportedly targeted the US Presidential election of 2016; the APT29 (Cozy Bear) group that is believed to have been behind the compromise of the US Democratic National Committee in 2015; a botnet believed to have been the largest on the internet at one time; targeted ransomware aimed at large businesses; and a banking Trojan considered to be so negatively impactful on victims that it was called out specially by the US Department of Homeland Security.

In addition to these known 'public' threats, the testers also generated a range of advanced targeted attacks using known malicious techniques, so creating files that were unique. These unknown files, which included malicious Microsoft Office and Javascript files, were included in the test so explore how **D-Client** handled malware of which it lacked prior knowledge.

The test comprised four main categories of attack: known, public malware campaigns; script-based targeted attacks (aka 'file-less') designed to avoid interacting with the hard disk of target systems; targeted attacks based on vulnerabilities in Microsoft Office file-formats and applications; and shellcode injection attacks, designed to exploit vulnerable software.

All of these approaches are commonly used to exploit computer systems with a view to gaining access and stealing information and/or causing damage.

**D-Client** successfully detected all of the public attacks and, additionally, protected the target systems from any ill effects, such as infection from ransomware and theft or destruction of data. Forensic examination of the targeted systems determined that no hidden issues were caused by the malware.

While the endpoint protection software detected and protected against all of the script-based targeted attacks, it generally did not clean up the malicious scripts. In all but two cases they were left intact on the target, providing a possible opportunity for victims to unwittingly copy the files to unprotected systems. Remediation of malicious scripts is now included in the latest version.

All of the Microsoft Office-related attacks and the shellcode injection attacks were scanned and detected and blocked from running. In total, **D-Client** protected against every one of the attacks launched in this test.

False positive testing, in which legitimate files are examined by security software, was used to ensure that **D-Client** was not configured to simply block every executed file. In this test **D-Client** generated no false positive results or any sub-optimum classifications of legitimate objects.

The test results demonstrate that **D-Client** was capable of both detecting and protecting against highly impactful threats launched in famous breaches and well-known malware campaigns, as well as more insidious and advanced targeted attacks.

## Appendices

### Appendix A: Terms Used

| TERM                 | MEANING  |
|----------------------|--|
| Blocked              | The attack was prevented from making any changes to the target.  |
| Complete Remediation | If a security product removes all significant traces of an attack, it has achieved complete remediation.   |
| Compromised          | The attack succeeded, resulting in malware running unhindered on the target. In the case of a targeted attack, the attacker was able to take remote control of the system and carry out a variety of tasks without hindrance.  |
| False Positive       | When a security product misclassifies a legitimate application or website as being malicious, it generates a 'false positive'.   |
| Neutralised          | The exploit or malware payload ran on the target but was subsequently removed.   |
| Target               | The test system that is protected by a security product.   |
| Threat               | A program or sequence of interactions with the target that is designed to take some level of unauthorised control of that target.  |
| Update               | Security vendors provide information to their products in an effort to keep abreast of the latest threats. These updates may be downloaded in bulk as one or more files, or requested individually and live over the internet. |

### Appendix B: FAQs

- The test was commissioned by **Deep Instinct**.
- The test was conducted in February 2019.
- The product was configured according to **Deep Instinct's** recommendations.
- Malicious URLs and legitimate applications were independently located and verified by SE Labs.
- Targeted attacks were selected and verified by SE Labs.
- Malicious and legitimate data was provided to Deep Instinct once the test was complete..
- SE Labs conducted this endpoint security test on physical PCs, not virtual machines.

### Appendix C: Product Versions

The table below shows the service's name as it was being marketed at the time of the test.

| PRODUCT VERSIONS |              |               |
|------------------|--------------|---------------|
| Provider         | Product Name | Build Version |
| Deep Instinct    | D-Client     | 2.2.1.5       |

The deep learning system was trained in August 2018.

**SE Labs Report Disclaimer**

1. The information contained in this report is subject to change and revision by SE Labs without notice.
2. SE Labs is under no obligation to update this report at any time.
3. SE Labs believes that the information contained within this report is accurate and reliable at the time of its publication, which can be found at the bottom of the contents page, but SE Labs does not guarantee this in any way.
4. All use of and any reliance on this report, or any information contained within this report, is solely at your own risk. SE Labs shall not be liable or responsible for any loss of profit (whether incurred directly or indirectly), any loss of goodwill or business reputation, any loss of data suffered, pure economic loss, cost of procurement of substitute goods or services, or other intangible loss, or any indirect, incidental, special or consequential loss, costs, damages, charges or expenses or exemplary damages arising his report in any way whatsoever.
5. The contents of this report does not constitute a recommendation, guarantee, endorsement or otherwise of any of the products listed, mentioned or tested.
6. The testing and subsequent results do not guarantee that there are no errors in the products, or that you will achieve the same or similar results. SE Labs does not guarantee in any way that the products will meet your expectations, requirements, specifications or needs.
7. Any trade marks, trade names, logos or images used in this report are the trade marks, trade names, logos or images of their respective owners.
8. The contents of this report are provided on an "AS IS" basis and accordingly SE Labs does not make any express or implied warranty or representation concerning its accuracy or completeness.