CYBERBIT
PROTECTING A NEW DIMENSION

Detection & Analysis of Dridex with CyberShield AnD for IT

Introduction

Dridex has been one of most notable threats since its appearance in late 2014 and throughout 2015. A strain of malware evolving from the infamous Zeus family, it is designed to steal personal banking information and credentials. Its operators seem to focus on small and medium-sized organizations as targets. The attack is said to be responsible for the theft of over $50 million, out of which $30 million stolen from UK accounts alone. A recent arrest by the FBI of Moldovan national Andrey Ghinku had contributed little to stopping the threat. The gang behind Dridex is believed to have links to similar cybercrime gangs, such as the so called 'Business Club' behind GameOver Zeus, and we believe that the experience and lessons gained by previous activities allow Dridex authors and affiliates to keep their infrastructure alive and to stay active and dangerous.

Part of Dridex's robustness is attributed to its polymorphism and obfuscation layers, which allow the constant generation of new variants for each campaign. These usually go undetected under most AV engines

While analyzing Dridex variants, we managed to uncover for the first time its advanced and sophisticated persistency mechanism, which provides the understanding of why it is so hard to detect. This persistency mechanism is described in details later in the report.

The following report provides the details of a dynamic behavioral analysis over several samples of Dridex, focusing mainly on its infection and persistency methods.

CYBERBIT COMMERCIAL SOLUTIONS Ltd.
22 Zarhin st. P.O.B. 4410, Ra'anana 4310602, Israel
Tel: 972-9-8864525, Fax: 972-9-8864556
cyber.sales@elbitsystems.com | www.cyberbitc.com

Executive Summary

Dridex produces a set of suspicious behavioral signatures that allow us to follow and map the way it operates, and understand its infection and persistency methods:

1 Dridex uses Office macros to download a malicious payload and execute it.

2 The malicious file injects code into Explorer.

3 The original file is erased from the disk.

4 The malware writes itself back to the disk only when the system begins to shut down.

5 Dridex changes Windows Firewall settings in order to allow the future remote communications via Explorer.

6 While operating in the Explorer window - Dridex steals bank accounts credentials that will allow its operators to legitimately connect to the users' bank accounts

Dridex malware is rich with techniques that make it harder to analyze - statically or dynamically, manually or automatically. They also make static signature detection dramatically less effective. These techniques include redundant opcodes, code snippets with no real functionality, and replacing necessary and important code functions with its own (GetProcAddress).

Finally, Dridex's sophisticated and stealthy persistency technique, fully revealed here for the first time, allows it to hardly leave any footprint on the file system and registry, making it almost impossible to detect. Dridex replaces the Window Procedure of Explorer with its own malicious function, awaiting system shutdown messages. Once a shutdown indication is received, Dridex writes itself into the disk, in order to assure it is executed again when the computer is restarted.

When the computer is restarted, Dridex code is one of the first things to be reloaded. Once the malicious code is again injected into Explorer, Dridex code is deleted from the disk, and again resides in memory alone.

This tactic assures Dridex can hardly be discovered, detected, or removed. Only after fully understating this persistency model, can organizations be capable of detecting and removing this malware.

CYBERBIT COMMERCIAL SOLUTIONS Ltd.
22 Zarhin st. P.O.B. 4410, Ra'anana 4310602, Israel
Tel: 972-9-8864525, Fax: 972-9-8864556
cyber.sales@elbitsystems.com | www.cyberbitc.com

Behavioral Analysis

A lot was written about Dridex's usage of Office macros as an infection method, contributing to the general rise in popularity of Office documents as an attack surface. But how does Dridex continue to fully infect the system and gain sufficient capabilities?

An execution of a typical Dridex variant produces the following behavioral signatures:

Program acts as a Dropper (winword.exe)

Office program spawns suspicious process (winword.exe)

Office program creates suspicious network connection (winword.exe)

Dropped executable has suspicious attributes in its file structure (Paame1.exe)

Program performs Code Injection (Paame1.exe)

Executable performs Self Deletion (Paame1.exe)

Possible UAC bypass by creating .sdb file (Explorer.exe)

Script performs Self Deletion (Eqf2WCUU.bat)

Program performs Code Injection (Explorer.exe)

Program creates PE with abnormal extension (Explorer.exe)

Program changes autorun settings (Explorer.exe)

CYBERBIT COMMERCIAL SOLUTIONS Ltd.
22 Zarhin st. P.O.B. 4410, Ra'anana 4310602, Israel
Tel: 972-9-8864525, Fax: 972-9-8864556
cyber.sales@elbitsystems.com | www.cyberbitc.com

A graph-view allows to see the connections between the involved entities and to better understand the general flow of malicious activity:

Figure 1 - graph view of Dridex analysis as presented by CyberShield AnD for IT

Dridex analysis graph allows us to better understand and map Dridex behavior:

1 As mentioned, Dridex uses Office macros containing VBScript in order to download a malicious payload and execute it.

2 The first payload, Paame1.exe, was found to have suspicious attributes in its file structure (and is therefore highlighted).

3 After connecting to a second host (to download a second payload), Paame1.exe injects code into Explorer.

4 After Explorer.exe deletes the first payload, Paame1.exe, there are no remains of the malware on the disk. The executable used for running after startup, 9b0.tmp, is created only when the system begins to shut down. Considering this, how can we be sure at this point that Dridex is running and affecting the system?

We can easily verify Dridex's presence with Process Explorer:

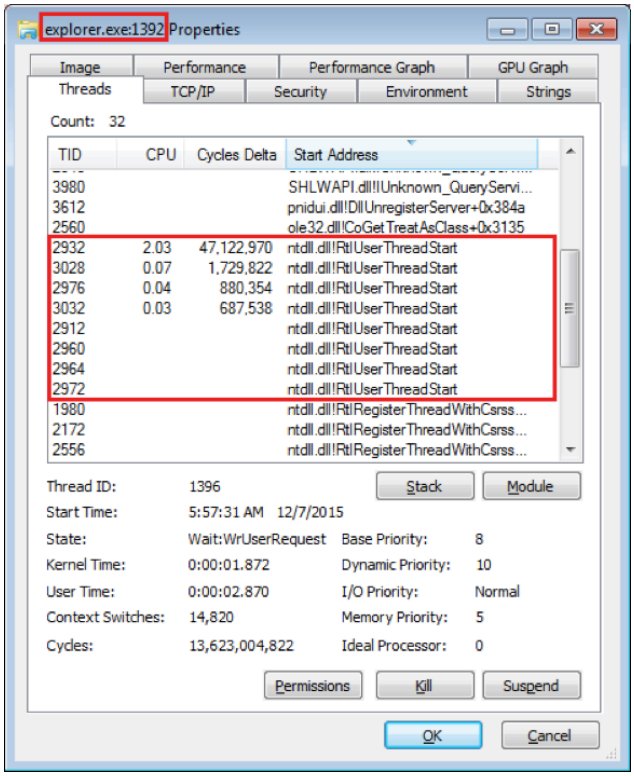

Figure 2 - Process Explorer indicating Dridex presence in the system

5 The creation of an .sdb file and spawning of iscsicli.exe is part of a known process for UAC Bypass for gaining administrator privileges. Other variants have been shown to use a more familiar method which uses sysprep.exe for UAC bypass. The common vulnerability all these Privilege Escalation exploit is the usage of auto-elevation programs.

6 In this case the Privilege Escalation was achieved in order to run Eqf2WCUU.bat with administrator rights. This caused the execution of a netsh process. Looking at netsh command line argument we can see it had the task of changing Windows Firewall settings, in order to allow the future remote communications via Explorer.

7 The bat file quickly deletes itself in order to hide its tracks. We were able to save it before its deletion and recover its contents:

8 The purpose of the code that Explorer injects into IExplore (or Chrome or Firefox) is to steal information from the user, especially credentials of banking accounts, according to a preconfigured list of online banking websites.

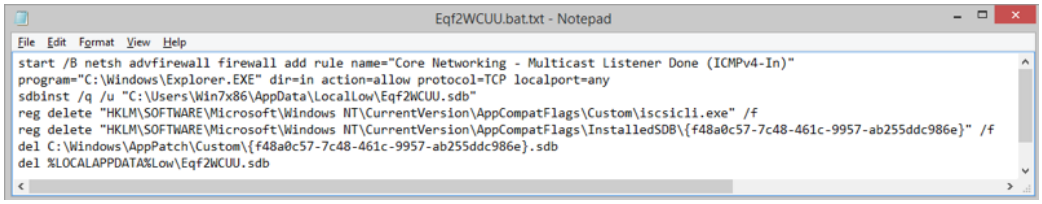

Figure 3 - deleted bat file contents

Anti-Analysis Methods

Additionally to the methods described above, Dridex is rich with many techniques designed to make it harder to analyze - statically or dynamically, manually or automatically. They also make static signature detection dramatically less effective.

1 Code Obfuscation #1:

The first payload that Winword downloads from the internet is usually a .net executable, originally written in C#. This C# contains many code snippets that look like the one below, which have no real functionality.

```
int num = 584;
if (num == 810)
{
    CodeCatchClauseCollection codeCatchClauseCollection = null;
    codeCatchClauseCollection.ToString();
    SmtpFailedRecipientsException ex = null;
    ex.GetHashCode();
    SetWin32ContextInIDispatchAttribute setWin32ContextInIDispatchAttribute = null;
    setWin32ContextInIDispatchAttribute.IsDefaultAttribute();
    AssemblyTrademarkAttribute assemblyTrademarkAttribute = null;
    assemblyTrademarkAttribute.IsDefaultAttribute();
    UnicodeEncoding unicodeEncoding = null;
    unicodeEncoding.GetPreamble();
}
```

Figure 4 - Dridex C# code snippets

2 Code Obfuscation #2:

When analyzing Dridex it's common to see redundant opcodes that were scrambled in.
For example, in the screenshot below we can see that the "not ecx" operation is followed by "lea ecx, [ebp+var_5c]". Since ecx is not being read between them, the first operation is useless. A closer look will reveal that the "imul" and "xchg" opcodes are also redundent.

```
00000AFA F7 D1          not     ecx             ; One's Complement Negation
00000AFC 6A 00          push    0
00000AFE 0F AF D6       imul    edx, esi        ; Signed Multiply
00000B01 6A 04          push    4
00000B03 87 DF          xchg    ebx, edi        ; Exchange Register/Memory with Register
00000B05 8D 4D A4       lea     ecx, [ebp+var_5C] ; Load Effective Address
```

Figure 5 - Dridex redundant opcodes

3 Code obfuscation #3:

Determining and inspecting the imported functions a program calls is very helpful when trying to understand its purpose and to decide whether it is malicious or not. The standard usage of Windows dynamic-link library, using import/export tables or GetProcAddress function, can be easily monitored. Dridex makes this kind of analysis harder, by implementing its own GetProcAddress. All calls to WinAPI within Dridex are done this way, as can be seen in this example:

```
00000B68
00000B68                loc_B68:                ; unsigned int
00000B68 68 45 98 BB F3 push    0F3BB9845h
00000B6D E8 A7 F4 FF FF call    get_dll_by_number ; Call Procedure
00000B72 68 55 C3 55 89 push    8955C355h
00000B77 50             push    eax
00000B78 E8 8B F5 FF FF call    get_winapi_function_handler ; Call Procedure
00000B7D FF D0          call    eax             ; Indirect Call Near Procedure
```

Figure 6 - Dridex designated GetProcAddress function

CYBERBIT COMMERCIAL SOLUTIONS Ltd.
22 Zarhin st. P.O.B. 4410, Ra'anana 4310602, Israel
Tel: 972-9-8864525, Fax: 972-9-8864556
cyber.sales@elbitsystems.com | www.cyberbitc.com

Persistency and Stealth

Several researchers have noticed that Dridex does not guarantee its own persistency until it absolutely has to, just a moment before shutdown. This tactic allows Dridex to hardly leave any footprint on the file system and registry, making it harder to detect and remove. How does it do it?

In Windows, every window has a Window Procedure. The Window Procedure is a function that is called in every event that occurs in the system in order to let the window respond to the event. For example, when the system is shutting down, it sends a message to all open windows so they can save their work.

Dridex uses this mechanism for its persistency. After injecting its malicious code into Explorer's process, Dridex replaces the Window Procedure of Explorer with its own malicious function. This is achieved by calling the SetWindowsLong WinAPI function. For every message the malicious function receives it calls the original function, except for the messages WM_QUERYENDSESSION and WM_ENDSESSION. These messages are sent when the system is shutting down. Once one of them is received, Dridex knows that the system is shutting down and that it should write itself into the registry autorun key, in order to make sure it is executed again when the computer is restarted.

The malicious Window Procedure function is shown here:

```
int __stdcall new_window_proc(int a1, int uMsg, int a3, int a4)
{
  if ( uMsg == 0x11 || uMsg == 0x16 )
  {
    constructor((int)&v13);
    sub_5CE64E3(97586560, 1);
    v4 = sub_5CE66EF(&v19, &v9, 97586668);
    sub_5CE593F(&v15, *(_DWORD *)(v4 + 4));
    string_destructor((int)&v9);
    sub_5CE5907(&v9);
    sub_5CE5907(&v7);
    v8(v16, 97566704, 0, v10);
    sub_5CE6987(&v17);
    v5 = *(_DWORD *)(dword_5D18B34 + 1040);
    sub_5CE6AB7(*(_DWORD *)(dword_5D18B34 + 1036));
    if ( v18 && v18 != -1 )
    {
      CloseHandleImplementatio(v18);
      v18 = 0;
    }
    sub_5CE59CA();
    v8(v10, v8, *(_DWORD *)&v7);
    sub_5CE40CD(&v17, v8);
    sub_5CE416A();
    string_destructor((int)&v17);
    sub_5CE59CA();
    sub_5CE59CA();
    sub_5CE59CA();
    sub_5CDB378();
    sub_5CE64E3(97586676, 2);
    strlen2(97586724, (int)&v11);
    sub_5CE4288(v14);
    sub_5CE4288(97586740);
    calls_set_regvalue(&v19, v12);
    SetEventStub(shutdown_event);
    string_destructor((int)&v11);
    sub_5CDB378();
    string_destructor((int)&v13);
  }
  return v8(original_window_proc, a1, uMsg, a3, a4);
}
```

Figure 7 - Dridex malicious Windows Procedures function

When the computer is restarted, Explorer reads the autorun registry keys and runs the programs they specify. One of the first things that Dridex's malware does when running is to delete its autorun key and the executable file that contains its code. The malicious code is again injected into Explorer, and now resides in memory alone.

Looking at the actions Dridex performs just before shutdown and immediately after startup, we can conclude that Dridex's traces on the disk and registry exist only for a very short period of time, in a timeframe which is also harder to monitor.

CYBERBIT
PROTECTING A NEW DIMENSION

Summary

Dridex malware, affecting mainly small and medium-sized organizations in order to steal personal banking information and credentials, is already accountable for over 50$ million financial damage. Its robustness is attributed to its polymorphism and obfuscation layers, which allow the constant generation of new variants for each campaign, undetected by most AV engines.

For the first time, in this report, we revealed Dridex's sophisticated persistency mechanism, which allows it to remain uncovered and undetected due to its mode of operation.

The report provided a detailed dynamic behavioral analysis of the Dridex malware, focusing on its infection process and persistency mechanism, in order to allow organizations to understand how to detect and remove it prior large scale damage is obtained.

The analysis was conducted by CYBERBIT's dedicated malware research expert team, which analyzes malwares and security threats in order to enrich analyses methods and algorithms in CyberShield AnD for IT

About CYBERBIT

Security threats in the 21st century are more complex, more sophisticated, and stealthier than ever - with an estimated 70%-90% of malwares which are unique to a specific targeted organization, managing to bypass traditional security tools.

Enterprises just can't afford to keep using technologies and methods that don't work.

We in CYBERBIT understand that in order to detect and respond fast and efficiently to advanced unknown threats organizational security has to be changed. Detection and response cycles must become optimal and short, leveraging granular information pieces as well as past knowledge, automating processes and capabilities, and allowing the organization to be agile, alert and prompt.

Our solutions aim to empower your team by providing them a different level of detection, response, forensics and mitigation capabilities and allowing them to operate fast, efficiently and accurately.

CYBERBIT's products collect and analyze pieces of information in greater depth and context over time and space, and provide ad-hoc forensics and response capabilities, for both IT and SCADA networks, while assuring minimum time for mitigation, remediation and response.

CYBERBIT believes that technology is leveraged by skilled, updated and competent personnel, and supplies live hands-on training that keeps your team efficient and slick.

CyberShield integrated cyber security suite constitutes of:

- **Endpoint Detection and Response (AnD for IT)**
- **SCADA Detection and Response (AnD for SCADA)**
- **Security Incident Response Platform (MnR)**
- **Security Training and Simulation (TnS)**

CYBERBIT COMMERCIAL SOLUTIONS Ltd.
22 Zarhin st. P.O.B. 4410, Ra'anana 4310602, Israel
Tel: 972-9-8864525, Fax: 972-9-8864556
cyber.sales@elbitsystems.com | www.cyberbitc.com

AnD for IT

AnD for IT is an endpoint detection and response solution built of strong and wide-spread kernel-level sensors; big data analytics for enhanced detection, data enrichment and forensics; and an incident response application to facilitate fast response execution.

AnD for IT's detection capabilities use behavioral and statistical analytics to detect advanced, targeted and unknown threats, over time and space, and identify their lateral movement and propagation, incorporating historical data and threat intelligence.

The big data infrastructure also provides AnD with advanced and ad-hoc forensics capabilities, which, combined with its high-end incident response application, empower the analysts and maximize manpower efficiency.

AnD's analytics are constantly enriched and updated by CYBERBIT's dedicated malware research experts team.

MD5s:

AB619931EBF56EE0137548F18209F38B
D7113159AC45B5958AD69D33B066529D
AC1D437E08BFE27942256DA9E1EE1293
1DE43352E20E7A34193C9DC7F7FB46FE
44CBD39C6581342252CE1CDD238B2975
CCD94E452B35F8820B88D1E5856E8196
27E7A76A691DC562B30DA8D98014D686
071B380D6B422DD83F14FA0A3BCEB347
CA8FDF844DF402076D37E85A74FA485B
FEA3AB857813C0D65CD0B6B6233A834B
DF2A0FCE92A362FA1D893B8F8B6F4629
666B2121CFB7871CD1354B08D51A36E4

References

http://www.theguardian.com/technology/2015/oct/14/what-is-dridex-how-can-i-stay-safe

http://www.securityweek.com/dridex-still-active-after-takedown-attempt

http://www.forbes.com/sites/thomasbrewster/2015/10/13/dridex-botnet-takedown/

http://stopmalvertising.com/malware-reports/analysis-of-dridex-cridex-feodo-bugat.html

http://blog.dynamoo.com/

CYBERBIT COMMERCIAL SOLUTIONS Ltd.
22 Zarhin st. P.O.B. 4410, Ra'anana 4310602, Israel
Tel: 972-9-8864525, Fax: 972-9-8864556
cyber.sales@elbitsystems.com | www.cyberbitc.com