Ursnif incident report

* **Executive Summary**: State in simple, direct terms what happened (when, who, what).
* **Details**: Details of the victim (hostname, IP address, MAC address, User account name,…etc).

# **Executive Summary**

Ursnif/Gozi malware, is worldwide trojan which is widely spread as a banking malware in 2000. The traffic data is at 2019-12-06. It collects data about the victims including their keystrokes and browsing activities. So, it is able to steal bank account details, credit card data, and login credentials.

Gozi was developed by **Nikita Kurmin**, and he borrowed code from Ursnif aka Snifula, a spyware developed by **Alexey Ivanov**. Gozi v1.0 often is classified as Ursnif. It became available in GitHub in 2015, so other developers can extend its functionality easily.

It performs data gathering through malicious phishing /spam campaigns effectively, but also can be spread using USB flash drives. The email/spam contains .zip attachment of type Microsoft office document (Such as Excel) that contains instructions to the victim to enable a macro (a single instruction that expands automatically into a set of instructions to perform a particular task).

The email is sent as if it is from the manager (with the manager signature), so that victims are more likely to open the file. The email includes a password that is required to open the file. Once opened, victim will have the file that contains URL. From that URL, the victim will have DLL downloaded on its machine, and malware will spread to infect the system.

# **Details of the victim**

# Victim’s Details

**Hostname** Smithers-PC **Mac Address** 00:08:02:1c:47:ae

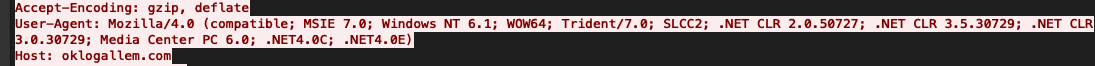
**IP Addresses** 10.11.12.101 **Public IP Addresses** 173.166.146.112

Graphical user interface, application

Description automatically generated

Graphical user interface, text

Description automatically generated with medium confidence

****

Text

Description automatically generated

\*User account is unavailable for the given the pcap files.

**Indicators of Compromise (IOCs)**

Text

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| IP address | Domain | Port | URL |
| 80.85.159.236 | oklogallem.com | 80 | <http://oklogallem.com/zepoli/ironak.php?l=luntsu1.cab> |
| 194.87.147.244 | kh2714ldb.com | 80 | <http://kh2714ldb.com/images/58HuD8VcxhOH06K/eUWS28C7Jfyw4oHXzL/_2FjxyFTs/2mwSlPW_2FV4rSxAQZq3/WEnEAO7KAF1nmNFHyd2/OPs9FAmiyy6Rzf_2FtqcEl/a_2Bn2XZvVjEI/OFh1gdQL/Ci2LaQBKQGuvqMSky2OytwY/gXim0rEnHt/qqtju0TWOjD0isRRc/_2BagK_2/B.avi> |
| 208.67.222.222 | resolver1.opendns.com | 53 | Not Applicable. |

Table

Description automatically generated with low confidence

|  |  |  |  |
| --- | --- | --- | --- |
| IP address | Domain | Port | URL |
| 85.143.219.95 | s9971kbjjessie.com | 443 | Encrypted |
| 124.217.255.96 | startuptshirt.my | 80 | <http://startuptshirt.my/wp-content/uploads/2019/11/jjasndeqw.rar> |
| 94.140.114.6 | Unavailable | 443 | Encrypted |
| 5.61.34.51 | Unavailable | 443 | Encrypted |

* SHA256 hashes of malware binaries that is extracted from the pcap

Text

Description automatically generated

This URL ending in .rar returned follow-up malware. However, this follow-up malware is encoded/encrypted when sent over the network. The binary decoded on the infected Windows host, which is not seen in the infection traffic, so we cannot export a copy of the follow-up malware from the pcap, and have its SHA256. But based on the post-infection traffic, the type of malware was sent to the Ursnif-infected host is Dridex.

**Technical Description**

General infection chain is as follow:

## Arrives as an office document attachment

## User tricked into opening document and executed malicious macro

## Users download malicious DLL

## DLL DLL is executed

## Malware steal data and credentials

## Victim‘s computer connect to remote server

## Remote server able to use backdoor commands

We can note the sequence of events:

* HTTP GET request that returns an initial Ursnif binary to oklogallem.com. (80.85.159.236) Port 80 [Stage#1 Recon]
* HTTP GET requests caused by the initial Ursnif binary, including decoy URLs to kh2714ldb.com before the infection becomes persistent.(194.87.147.244) Port 80 [Stage#2 Delivery]
* HTTPS traffic after Ursnif is persistent in the Windows registry, Ursnif causes HTTPS traffic to s9971kbjjessie.com [Stage#3 Exploit]
* HTTP GET request for follow-up malware HTTP GET request to startuptshirt.my [Stage#4 Install]
* Post-infection activity from the follow-up malware [Stage#5 C&C],[Stage#6 Exfiltrate]

# Impact

Since it causes information theft as the main consequence, this will impact victims financial data such as loss includes stealing bank and digital wallets and cryptocurrency information. It will also impact and Violate the victims’ privacy such as gathers victims’ credentials, logs keystroke and steals user data.

## Recommendation to prevent Ursnif

There are several solutions, include the following:

## Email Protection

## URL Protection

## Network Pattern

## File Detection

## Predictive Learning

## Advance Threat Scan Engine

## References:

* [https://success.trendmicro.com/solution/000283513#:~:text=Ursnif%20malware%2C%20also%20known%20as,most%20widely%20spread%20banking%20Trojan.&text=Ursnif%20malware%20is%20effectively%20delivered,the%20user%20to%20enable%20macro](https://success.trendmicro.com/solution/000283513" \l ":~:text=Ursnif%20malware%2C%20also%20known%20as,most%20widely%20spread%20banking%20Trojan.&text=Ursnif%20malware%20is%20effectively%20delivered,the%20user%20to%20enable%20macro)
* <https://malpedia.caad.fkie.fraunhofer.de/details/win.gozi#:~:text=It%20was%20offered%20as%20a,classified%20as%20Ursnif%20aka%20Snifula>
* <https://unit42.paloaltonetworks.com/using-wireshark-identifying-hosts-and-users/>