Cafe Cracks: Attacks on Unsecured Wireless Networks

Paul Moceri (paul.moceri@gmail.com) and Troy Ruths (manlytwah@gmail.com)

Abstract

Mobile users demand high connectivity in today's world, often at the price of security. Requiring Internet access at the airport, public buildings, and restaurants, users will easily sacrifice a secure connection for a fast and reliable one. By broadcasting rogue access points at these compromising locations, crackers can launch effective Man-in-the-Middle attacks. Our developed crack, Cafe Crack, provides a platform built from open source software for deploying rogue access points and sophisticated Man-in-the-Middle attacks. Built around the Untangle Server software, Cafe Crack allows the hacker to dynamically measure, monitor and redirect network traffic. This paper will provide an example of DNS spoofing using the Cafe Crack platform and then provide simple and effective protection techniques against harmful rogue AP attacks.

Keywords

Spoofing, phishing, DNS, DNSSEC, VPN, Public VPN, Untangle, Man-in-the-Middle, wireless, 802.11, Rogue AP, Open Source, Evil Twin

---

Table of Contents

---

1 Introduction

Internet users have grown accustomed to ubiquitous connectivity. Mobile users, in particular, constantly look for open networks to use. Often times, users end up connecting to insecure networks without taking the necessary security measures. With the proliferation of wireless networks in public places, this is becoming a greater concern.

This project demonstrates how a cracker could easily spoof a public wireless access point (AP) and use it to collect private information from an individual. We will show all of the steps necessarily to setup a mobile platform, called Cafe Crack, to launch such attacks. Then, we will describe ways to protect against these attacks when using unsecured and distrusted networks.

1.1 Rogue AP Attack

The rogue AP attack is successful since it capitalizes on the incompetency of the user rather than technology. By broadcasting a fake wireless network in a trusted area, rogue APs lure victims into connecting to the fake network and sharing sensitive information. Also known as the Evil Twin attack, a hacker hijacks a trusted wireless network by force or proximity, stealing users from the original network. Users connect and conduct their normal affairs on the Internet, unwittingly supplying the hacker with usernames, passwords, credit card numbers and other private information through phishing sites.

1.2 Airsnarf

Airsnarf is a rogue AP setup utility designed to phish for user information using a network log in page and Domain Name System (DNS) spoofing [Airsnarf
]. The tool is comprised of TreeWalk, SoftAP, Apache, ActivePerl, and several scripts all running on Linux. Airsnarf demonstrates the vulnerability in wireless networks by exploiting the rouge AP and Evil Twin attacks; however the software is no longer supported. By broadcasting a competing wireless network, Airsnarf phishes for username and passwords from victims, in addition to credit card information taken on the network log in page. By serving as the DNS server, the software could also be used to redirect victims to false secure pages in order to steal sensitive information. Our developed crack builds on the ideology of Airsnarf, and enhances the capabilities of the cracker by using a software network gateway, Untangle. This means the Man-in-the-Middle attacks can be more sophisticated and diverse using the same Cafe Crack platform.

1.3 Detecting Rogue APs

Currently, there are hardware and software solutions for detecting rouge APs manually and automatically. The easiest form of detection is to police the public area for untrusted wireless networks by matching a network's Service Set Identifier (SSID) with the MAC address of the access point; however, this does not provide a real solution since this would require constant human observation. A popular package that can help with this type of detection is NetStumbler [NetStumbler]. An automated solution is to use hardware probes that continuously monitor network traffic. This, though, tends to be costly and can be troublesome for areas with overlapping Wi-Fi hotspots. In the end, the easiest and most effective way to protect against rogue APs is self-protection, since policing the entire wireless network range is difficult and costly.

Back to Table of Contents

---

2 Cafe Crack

In many public places, such as Cafes, coffee shops, airports and hotels, wireless networks are broadcasted without any security because of the difficulty of providing security to transient users. These places are more concerned with providing an easy to access service for their customer than a secure service. However, this causes two major gaps in security. First, the user has no way of authenticating the wireless network to know if it is a trustworthy network or an attacker. Second, traffic is broadcasted in the clear for anyone to capture. These are major problems because many users intentionally seek out unsecured networks for free Internet access.

Cafe Crack is a platform we built using existing open-source software in order to capitalize on these security problems and take advantage of users who seek out these networks. By spoofing an unsecured, public wireless network, Cafe Crack entices victims to turnover personal information without realizing an attack has been made. In the series of Cafe Crack attacks, the attacker is able to obtain private information from the user because of his position as a man-in-the-middle.

2.1 Attack Machine

In order to spoof an unsecured, public wireless network, the attacker must be mobile and unassuming. If the attacker is not mobile, it would be easy to find the spoofed wireless network since it would consistently occur in the same area, despite identity changes to the wireless broadcast. Also, if suspicions arise over the attacker, he can easily get up and leave. Finally, the attacker must not be noticeable in a public location, since this could put potential victims on edge.

Therefore the ideal attack machine is a laptop equipped with two wireless cards. One wireless card is used to access an existing, legitimate wireless network that will be used to provide Internet access to victims. The other wireless card is used to broadcast an unsecured wireless network that victims will connect. Ideally, this second wireless card must support AP mode. This means that the card looks like a base station to other wireless clients rather than a peer-to-peer ad-hoc network. In practice, this is the most difficult part of the attack. Few wireless cards support AP mode and even fewer have available drivers. However, there are plenty resources for configuring a network card in AP mode, particularly on Linux [Tzanidakis06]. We ended up running our attack on Windows which only allowed us to broadcast an ad-hoc network instead of an AP-based network. This still allows for the same attack. However, using AP mode would make an actual attack even more convincing to the unsuspecting user.

To eliminate the need for any additional hardware, the rest of Cafe Crack has been developed in software, allowing the attacker to appear as just another wireless user with only a laptop. The basic configuration and requirements of the Cafe Crack are detailed in Figure 1. The attack machine connects to the Internet through a public wireless network and then broadcasts a spoofed wireless network. In order to provide convincing Internet experience and to complete the attack, the attacking computer runs several software components depicted below. Each of these components are described in the following sections.

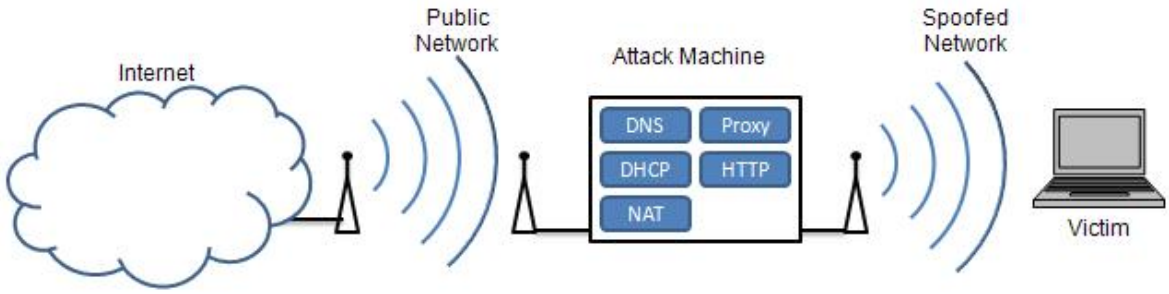

Figure 1 Diagram showing on overview of Cafe Crack. Note there are two wireless networks: one legitimate connection that the attacker uses to connect to Internet and also the spoofed network that victims connect to.

2.2 Components

The attack machine must run a variety of services to entice users, redirect their requests, and phish for personal information. In order to attract users, the attacker must broadcast a valid wireless access point capable of providing connectivity to the Internet, which requires Dynamic Host Configuration Protocol (DHCP) service and Network

Address Translation (NAT). These services allow the network to assign IP addresses and route traffic, respectively. The attacking network must also provide DNS service to users so that they can use the Internet. The network accomplishes this by providing the IP address of a DNS resolution server to the client as part of DHCP configuration. This address can be for a DNS server from the ISP or another third-party. However, for attack purposed, the attack machine runs its own DNS server and supplies its address to clients using DHCP. This allows the attacker to redirect users to phishing sites using DNS.

It is possible that users will have some of their own network parameters, such as DNS server, manually configured. To still perform DNS attacks the attack machine will need to be able to perform packet redirects as part of its routing capabilities. For example, since all DNS request traffic runs on User Datagram Protocol (UDP) port 53, the attack machine can easily redirect this traffic, regardless of the destination, to its own DNS server. Finally, to phish for private information, targeted sites will be hosted by a web server, designed to closely match the experience when visiting the real page. The DNS service will supply the web servers IP address when redirect for a phishing site. In order to change targeted sites, the attacker only needs to edit the DNS and web server. These changes can happen during uptime, so that the targeted sites may be dynamic and change based on time of day or usage. Table 1 summarizes the components of the Cafe Crack platform and the roles they perform in the attack.

Table 1 Summary of the various components of Cafe Crack and their role in the attack

Component	Role in attack
DHCP Server	Dynamically assigns IP addresses to network users so that they can use the network
DNS Server	Provides domain name resolution for user so that they can use the Internet and so attacker can redirect users
NAT	Provides address translation so that multiple users can be connected to the network
Web Server	Allows the attacker to host phishing web sites

Back to Table of Contents

---

3 Implementation

Cafe Crack uses Untangle, an open source network gateway and the Apache web server to accomplish several attacks. Cafe Crack is intended to show how easy it is to perform wireless attacks using only free, easy-to-use software. We chose to use Untangle since it was recently released (June 2007) under the GPL v2 and provides an award-winning software network gateway [Prince07]. Untangle allows for configurable DNS, DHCP, routing, and network redirection services. Each section below will explain the configuration for each of the necessary components provided by Untangle or Apache.

Untangle is distributed as an application running on its own Linux distribution. It comes as a complete installation or it can also be run as a virtual machine. We chose to run it as a virtual machine using VMware Server on Windows [VMware]. The installation of Untangle on VMware Server is well documented on the Untangle Wiki site [UntangleWiki]. Since, we used a virtual machine for Untangle, we were able to take advantage of the host machine to run other aspects of the attack. In our case, we ran a web server on the host machine to server phishing site. However, we could have also used another virtual server or even an external web server for the phishing site. The implemented example will use both an internal and external IP address, but ultimately the web server needs to resolve to an IP address that the DNS service can redirect to.

3.1 Routing

The first step in configuring Untangle for our attack was to set up routing using NAT. NAT will translate the internal network IP addresses to an external address so that multiple users will be able to connect through the same IP address on the attack machine's Internet connection. To do this, we enabled NAT and set the internal address and subnet mask

to 192.168.0.1 and 255.255.0.0. This address and subnet can be any subnet from one of the reserved private address block but should not conflict with the subnet that will be providing Internet access for the attack machine. The external address is set for dynamic configuration. This is important because the external IP address will be provided by the network that the attacker uses to connect to the Internet. Figure 2 shows the routing configuration including internal and external IP addresses in Untangle.

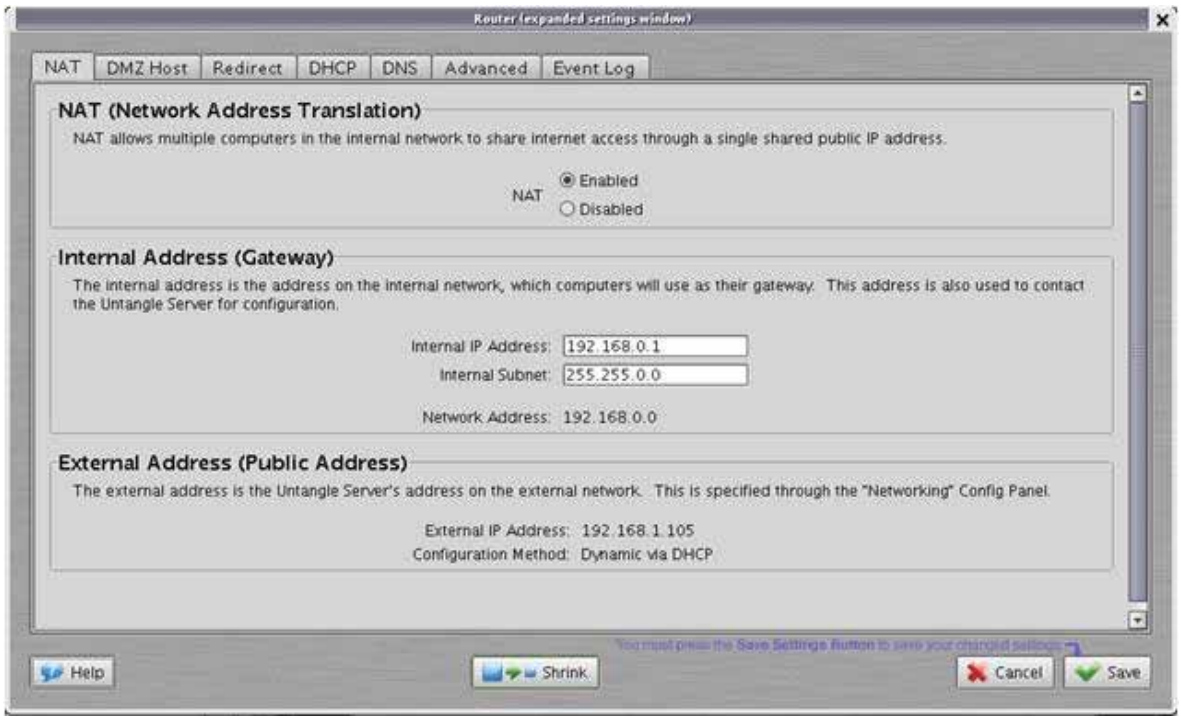

Figure 2 A screenshot of Untangle's routing configuration for Cafe Crack.

3.2 DHCP

Using DNS to our advantage is an important part of this attack. Thus it is important that we use DHCP to configure users the way we want. DHCP is enabled with an IP address range that matches the internal subnet creating above in the routing configuration. As seen in Figure 3, in this example we used 192.168.0.100 to 192.168.0.200. This range can change depending on the subnet selected and the number of available IP addresses desired.

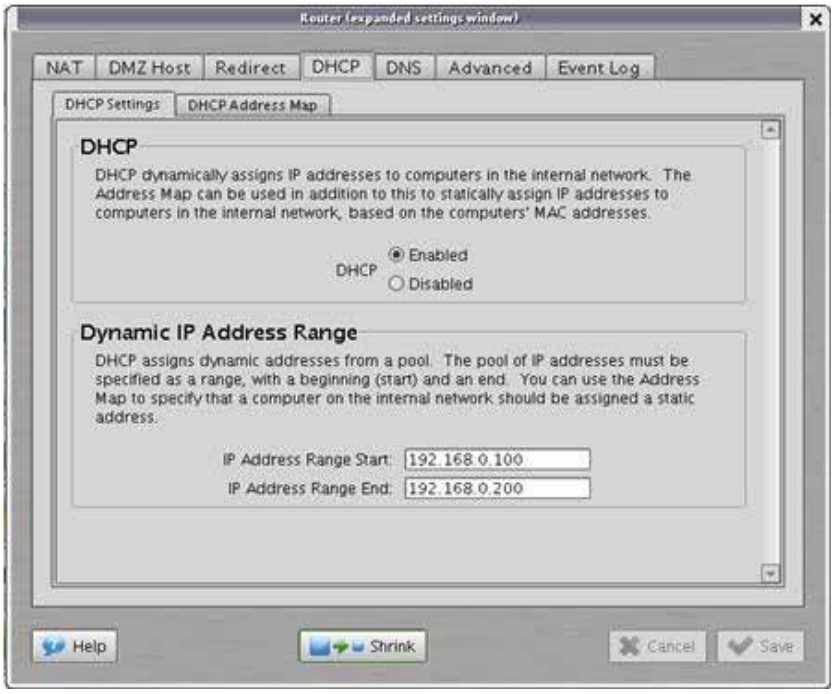

Figure 3 A screenshot of the DHCP setting in Untangle.

Also, we need a static IP address for the physical machine, since we will be using it as a web server. To accomplish this, a row is added to the DHCP Address Map for the Media Access Control (MAC) address of the physical attack machine. Figure 4 show this configuration in Untangle. In this example, the physical machine has been assigned the IP address 192.168.0.99.

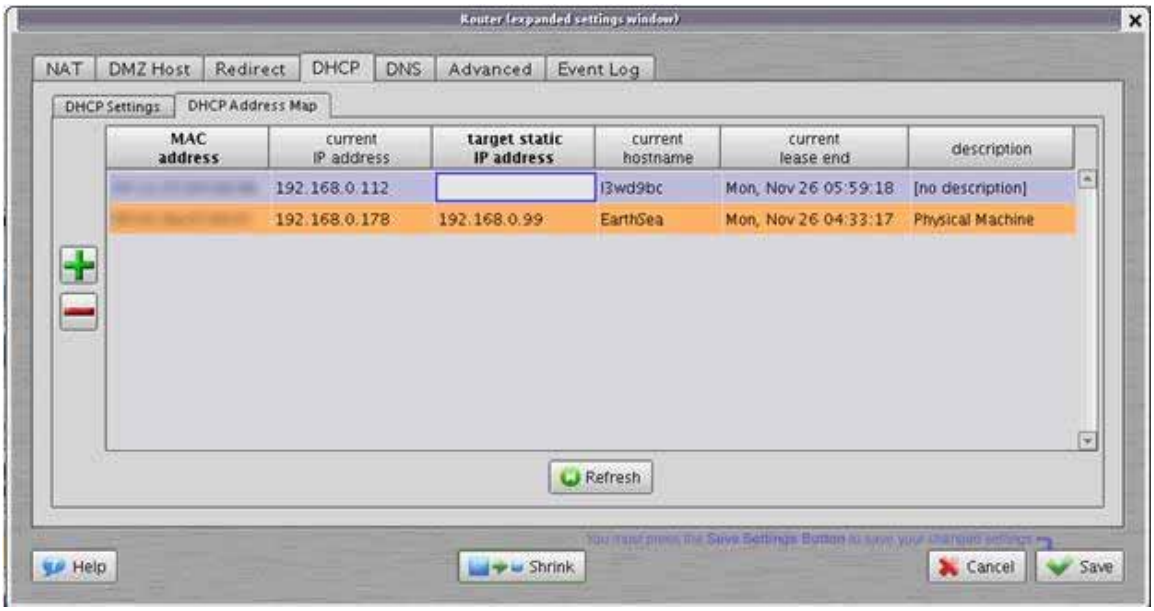

Figure 4 A screenshot of the statics IP configuration of the Untangle server.

3.3 DNS

The DNS service redirects the victim to the phishing page. On the DNS pane add the targeted sites to the list and the IP address where the server is hosting the phishing page. Figure 5 show the DNS entries for the phishing sites. We have redirected *gmail.com*
to an internal phishing site on the Untangle server evident by the IP address of the Untangle internal interface, 192.168.0.99. We also demonstrate redirection to an external site. In this case, *citibank.com* is redirected to a site running on the public IP address 71.85.149.75. All other DNS requests that are not explicitly specified in this list will

automatically be forwarded by Untangle to the DNS servers supplied by the external network.

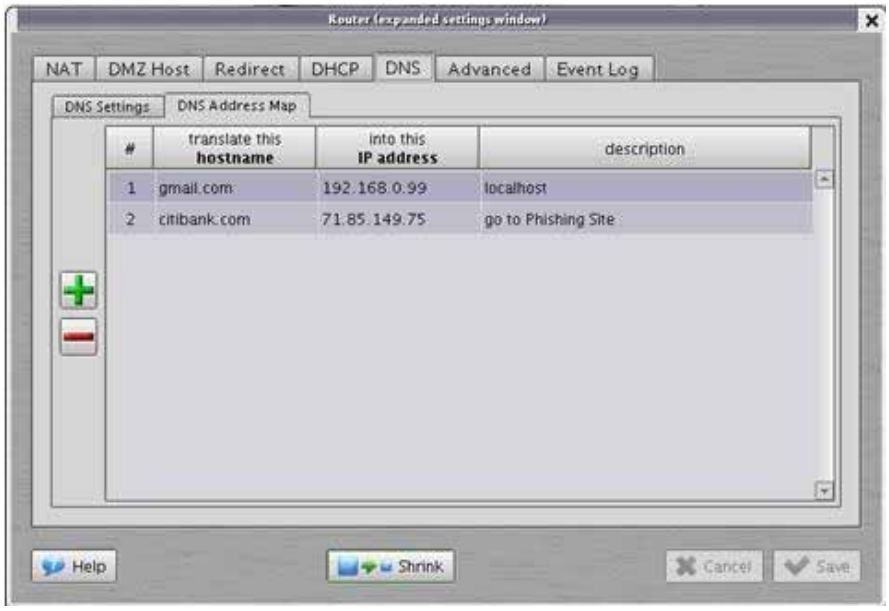

Figure 5 A screenshot of the DNS Address Map configuration in Untangle.

3.4 DNS Redirection

In order to redirect external DNS requests to the DNS service running on the Untangle server, we set up a port forwarding rule to forward all DNS requests to our Untangle server regardless of their destination. Since DNS requests use the well known UDP port 53, we can redirect such requests based on port number. Under the Redirect tab in Untangle, we created a new rule in "To Anywhere." Table 2 summarizes the configuration for this new rule.

Table 2 Forwarding rule for forwarding all DNS requests to our DNS server.

Traffic type	UDP
Source Interface	Internal
Destination Interface	External
Source Address	Any
Destination Address	Any
Source Port	Any
Destination Port	53
Redirect New Address	192.168.0.1 (IP of Untangle Server)
Redirect New Port	53

3.5 Webserver

The final necessary piece of this attack is a webserver to host the phishing site. In our example, we are running two different servers, one internally and one externally. On the attack machine, we are running Apache HTTP Server from the Apache Software Foundation [Apache]. The server runs on the host operating system of the attack machine and not on the virtual machine. The Apache server binds to an IP address on the internal network interface so that it is reachable by users on the spoofed network. For the external phishing site, we utilized Microsoft Windows built-in webserver, Internet Information Services (IIS), connected via cable modem to the Internet.

3.5.1 GMail Phishing Site

The first step in creating a convincing phishing site is to copy the original. In this case, we copied the GMail log in page and saved it as *ServiceLogin.html*. Next, we must reconfigure the page to redirect log in requests to a server side script, which will store the email and password information, and then forward the user onto the real GMail site. The real GMail site will prompt the user for the user name and password again, due to some sort of error. Since server errors as well as user typing errors happen with some frequency, the user will not be too alarmed at this request.

3.5.2 PHP Code

The first step in redirection is to mimic the long address of the GMail server request. We do this with a PHP redirect, shown in Figure 6.

```
header("Location:
ServiceLogin.html?service=mail&passive=true&rm=false&continue=https%3A%2F%2Fm
```

Figure 6 The index.php file redirects to a phishing page that replicates the long address query of the actual GMail site.

In *ServiceLogin.html*, we set the action property in the log in form to *phish.php*. Now, *phish.php* receives the log in information sent via POST. Figure 7 shares the code necessary to retrieve the log in variables. This code in phish.php retrieves the POST variables including username and password and redirects the user to the real GMail site. The user's log in information is hidden in the GET query for the header request.

```
$email = $_POST['Email'];
$pwd = $_POST['Passwd'];
header("Location:
https://www.google.com/accounts/ServiceLoginAuth?service=mail&Email=$email&Pa
```

Figure 7 PHP code from phish.php that captures users.

Once we have retrieved the log in information, it can be stored in a file or database for later exploitation. In this example, we just present the user with their user name and password in the redirection query visible in the browser's address bar. In a real attack, the code would forward the users on to the real Gmail log in page. This demonstrates the simplicity needed to design a convincing phishing site. For more complete source code, check out index.phps and phish.phps.

3.6 The Attacks

Several different security attacks are possible using our Cafe Crack platform. We have explicitly setup the machine to perform two DNS attacks. However, since all user traffic passes through our Untangle router, many additional man-in-the-middle attacks are possible.

3.6.1 Spoofed DNS

The basic attack using Cafe Crack is the spoofed DNS attack. An unsuspecting user receives dynamic configuration information including DNS servers from the attack machine. Then, as DNS requests are received by the attack machine, the DNS server either returns the IP address of a phishing site if it is one of the targeted spoofed sites or returns a legitimate IP address for sites that are not being spoofed.

3.6.2 DNS Redirection

DNS redirection is a more sophisticated version of the DNS spoofing. If a user has manually configured their own DNS servers, the spoofed configuration supplied by DHCP will not work and the user will continue to resolve correctly, so long as the manually configured server is trustworthy. With the redirection, all DNS packets are intercepted by Untangle and forwarded to the attacker's DNS service. The victim will then receive spoofed IP addresses for targeted

sites.

3.6.3 Other Attacks

The above two attacks both use DNS to direct users to spoofing sites. However, the Cafe Crack platform is able to perform many other attacks with various levels of sophistication.

Packet Sniffing
Since all traffic is routing through the Untangle server, we can simple capture all packets and search them for personal information such as usernames and passwords. However, this is a rather simple attack and can easily be thwarted by using encryption.

HTTP Proxy
Again, since all traffic passes through our router, we can once again manipulate it to obtain private information. For example, using an HTTP proxy server, we can remove SSL protection from websites. Some websites require SSL connections which would make packet sniffing useless. However, the Cafe Crack proxy server can play the role of the HTTPS client endpoint. The proxy server receives encrypted traffic from the website but then forwards it to the client unencrypted. The user will then reply in clear text as well, revealing private information. This satisfies the server's requirement for encryption but allows for unencrypted traffic with the user.

Back to Table of Contents

---

4 Protecting Yourself

Despite the privacy problems with unsecured wireless networks, it is still possible to use a public, unsecured wireless for safe network access. There are various steps a user can take to prevent these attacks. Several easy and free actions can protect against the simple attacks or detect when a user is vulnerable, while more sophisticated and costly actions can protect a user against all but the most sophisticated cryptographic cracks.

4.1 Securing DNS

The easiest way to protect against spoofing and phishing attacks is to simply ensure that DNS resolution can be trusted. If the attacker does not control name resolutions, it becomes much harder to trick a user into using a phishing site. With this control of DNS resolution, an attacker can replace any well know domain name with a phishing site. However, without access to the user's DNS, the attacker can only phish by using a domain name that is similar, but not the same, as a well known name. In this case, the attacker is relying on a user to not notice the difference between the actual domain name and the phishing site. An example of this would be using *www.cnn.com.somewhere.com* as a phishing site for those thinking they are going to *www.cnn.com*.

Phishing with similar addresses is a well known attack and most modern web browsers have built-in phishing protection. Firefox for example maintains a list of known phishing sites and prevents users from navigating to them [Firefox]. However, in a man-in-the-middle attack such as those in Cafe Crack, the attacker is able to maintain the actual domain name but substitute the IP address of the phishing site. This DNS spoofing is not detected by anti-phishing software such as that in Firefox because the legitimate site is not listed. DNS spoofing and many other DNS threats have been known for some time now and are described in *RFC3833: Threat Analysis of the Domain Name System* [RFC3833].

There are two ways to protect against such attacks. First, users could use a trusted third-party DNS service instead of the supplied DNS to provide some protection. Second, the move to secure DNS using DNS Security Extensions (DNSSEC) would provide safe DNS resolution

4.1.1 Third-Party DNS

Almost all networks provide DNS server addresses as part of dynamic host configuration (DHCP) and thus most users just use these servers. However, this provides an easy opportunity for an attacker to specify his own compromised DNS servers. To protect against this threat, users can instead use a trusted third-party DNS resolution service when on an unsecured network. There are several choices for alternative DNS servers, including free DNS services such as OpenDNS.

OpenDNS is a free, public available DNS service that provides not only DNS resolution but additional security features [OpenDNS]. OpenDNS provides extra security by blocking requests to known phishing and other malicious sites. To use this service, a user needs only to manually set their DNS servers in their computer's network settings. This is a onetime change because the OpenDNS service can be used for any Internet connect.

Other options for third-party DNS include using an employer's or a home Internet Service Provider's (ISP) DNS servers. Not all, but some companies will have public facing DNS servers which employees can use for DNS resolution regardless of how they are connected to the Internet. Also, some ISPs make their DNS resolution servers available to their customers globally. When available, a user should use one of these alternate DNS sources by manually configuring their computer's network connection with the IP addresses of these servers. However, though this adds some security, it is not free from attacks of its own.

4.1.2 DNS Security Extensions (DNSSEC)

The Domain Name System dates back to the beginnings of the Internet. Since the Internet was initially developed in a private forum, DNS, like many other protocols, has many insecurities that were not exploited until the Internet was widely used by the public. However, now that Internet use is almost ubiquitous, these vulnerabilities have surfaced. DNSSEC is an improvement to DNS that provides integrity and protects against source spoofing DNS attacks including the redirection attack described above [Haddad04]. Although DNSSEC solves many of the security problems with DNS, it is not widely used because of the large cost of transitioning from DNS. However, where available, DNSSEC protects against the DNS attacks in Cafe Crack and should be used by Internet users.

4.2 Require HTTPS

A great way to ensure privacy on unsecured networks is to require application layer security. When using the web, users should require all sites that send or receive personal information to use SSL. Most modern browsers provide a way to do this either natively or through extensions. Internet Explorer has built-in security zones which allow users to specify sites that should require HTTPS authentication. And, using extension, Firefox can automatically redirect websites to their secure version [Egilsson07].

4.3 VPN

A Virtual Private Network (VPN) connection is the best way to protect against attacks on an insecure or unknown network, wired or wireless. A VPN works by encapsulating network traffic in an encrypted tunnel to a secure endpoint. Traffic is both encrypted and authenticated, rendering an attacker helpless to read or modify traffic. Traffic could be modified by the attacker; however it could only be modified in a way unknown to the attacker and would automatically be detected by the user because of cryptographic integrity checks. VPN connections require a software VPN client on the user's machine and a VPN server on the Internet, beyond the unsecure access network. Figure 6 show a VPN tunnel through an insecure wireless link. Traffic is protected by the VPN until it reaches the VPN server and then forwarded on through the Internet.

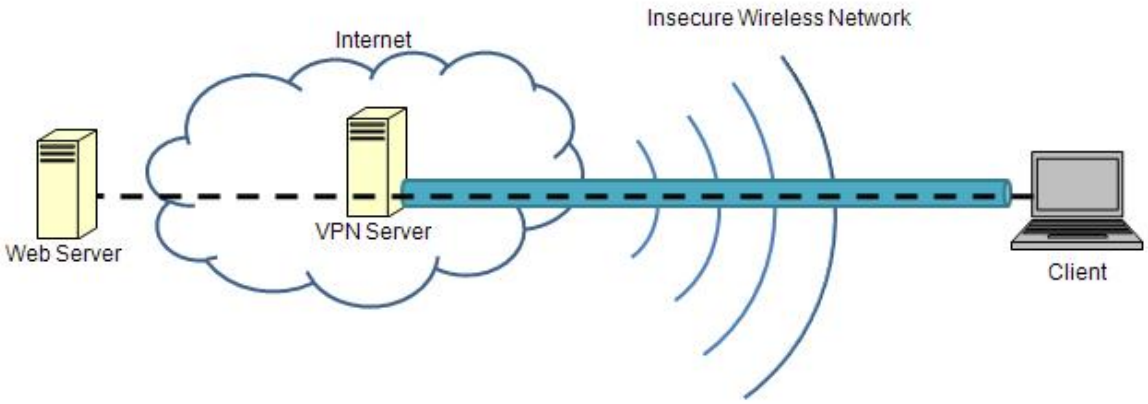

Figure 8 Overview of how VPNs provide security over an unsecured wireless link.

Users have more than once choice for where they can get their VPN service. VPN service can be accomplished either through private means or through public available VPN services.

4.3.1 Private VPN

Many companies, universities, and organizations provide their users with remote access to their private network via VPN service. This service provides security for all traffic between the user and organization's network over the public Internet. Thus, when connecting to an untrusted network, a user should use a VPN to secure all traffic on the unsecured network. Internet traffic should be routed to the VPN server and then forwarded into the Internet by the server. If VPN access is not provided by an employer, users could also setup VPN access using VPN tools on their own home Internet connection. That is, when away from home, the user could connect via VPN into their home and use their home Internet connect to ensure secure access.

4.3.2 Public VPN Service

For those who do not have access to a private VPN connection, there are also VPN services. These services provide a subscription to a VPN server that sits on the Internet and allows for secure communication on unsecured networks. An example of this type of service is PublicVPN.com [PublicVPN]. PublicVPN provides VPN access to the Internet using the VPN tools built into most operating systems. For example, Windows 2000 and XP provide a VPN client by default. For users without any other VPN options, this type of service is relatively inexpensive at only about $6 per month.

Back to Table of Contents

---

5 Conclusion

Because of the desire for continuous Internet connectivity, users often neglect security in favor of a free connection. This often means connecting to unsecured and untrusted wireless networks. Even worse, once connected to these unsecure networks, users often fail to take steps to protect themselves. In the paper, we have demonstrated an attack on the common user who connects to unsecured and untrusted wireless networks.

Our attack platform, Cafe Crack, broadcasts an unsecured wireless network with Internet connectivity to entice users to connect. Once connected, users' traffic is manipulated so that the attacker can obtain personal information from the users. While the primary method of attack used in this paper is DNS spoofing, Cafe Crack provides a platform for various other man-in-the-middle attacks. The simplicity of this crack amplifies its worth. Using only a laptop, the attacker can sit unassumingly in a public location to steal personal information. Perhaps the most alarming aspect of this demonstration is that it was accomplished with only a laptop and existing open-source software.

However, the good news is that it is just as easy to protect oneself against Man-in-the-Middle attacks on an unsecure

wireless connection. By using DNSSEC or VPN services, the user can bypass the attacker and keep their information secure. In the end, it is up to the user to be knowledgeable and safe around unsecure technology like public wireless.

Back to Table of Contents

References

The following references are roughly arranged in order of usefulness and relevance to the above paper.

[UntangleWiki] "Untangle Virtual Appliance on VMware" Untangle. Oct, 17, 2007.
http://wiki.untangle.com/index.php/Untangle_Virtual_Appliance_on_VMware
Wiki article describing how to run Untangle as a Virtual Server on VMWare.

[Haddad04] Haddad, Ibrahim and Gordon, David. "The Basics of DNSSEC" ONLamp.com, O'Reilly, October 14, 2006.
http://www.onlamp.com/pub/a/onlamp/2004/10/14/dnssec.html
Article that gives an overview of DNSSEC.

[RFC3963] Atkins, D. and Austein, R. "RFC3833: Threat Analysis of the Domain Name System," IETF, Network Working Group, August, 2004.
http://www.ietf.org/rfc/rfc3833.txt
Detailed analysis of the security threats in DNS.

[Untangle] "Untangle - The Open Source Network Gateway". Untangle.
http://www.untangle.com/
Untangle Home Page.

[Airsnarf] "Airsnarf" The Shmoo Group.
http://airsnarf.shmoo.com/
Homepage of the Airsnarf tool.

[Tzanidakis06] Tzanidakis, Manolis. "Creating a sSecure Linux-based Wireless Access Point" Linux.com. July 19, 2006.
http://www.linux.com/articles/55617
An article demonstrating how to setup a wireless network card in access point mode in Linux.

[Prince07]Prince, Brian. "Untangle Targets SMBs with Open-Source Security Platform". EWeek.com. June 26, 2007.
http://www.eweek.com/article2/0%2C1759%2C2151248%2C00.asp
A description of Untangle that appeared in a computing magazine.

[VMware] "VMware Server". VMware.
http://www.vmware.com/products/server/
VMware Server product page.

[OpenDNS] "OpenDNS | Providing A Safer And Faster Internet" OpenDNS.
http://www.opendns.com
OpenDNS website which includes overview of service and instructions for use.

[Apache] "The Apache HTTP Server Project" Apache Software Foundation.
http://httpd.apache.org/
Homepage of the Apache HTTP Server Project.

[Firefox] "Firefox Phishing Protection." Mozilla.
http://www.mozilla.com/en-US/firefox/phishing-protection/
Description of how Mozilla Firefox's built-in phishing protection works.

[PublicVPN] "PublicVPN.com" Access Nov 19, 2007.
http://www.publicvpn.com
Home page of Public VPN a subscription VPN service.

[NetStumbler] "Netstumbler.com"
http://www.netstumbler.com/
Homepage of the popular, free wireless network tool.

[Egilsson07] Egilsson, Einar. "Redirector :: Firefox Add-ons" Mozilla Software Foundation. October 5, 2007.
https://addons.mozilla.org/en-US/firefox/addon/5064
Product page for the Redirector Firefox extension by Einar Egilsson.

Back to Table of Contents

Acronyms

AP	Access Point
DNS	Domain Name System
DNSSEC	DNS Security Extensions
DHCP	Dynamic Host Configuration Protocol
GPL	GNU Public License
HTTP	Hypertext Transport Protocol
HTTPS	HTTP over SSL
IIS	Internet Information Services
IP	Internet Protocol
ISP	Internet Service Provider
MAC	Media Access Control
NAT	Network Address Translation
SSID	Service Set Identifier
SSL	Secure Sockets Layer
VPN	Virtual Private Network
UDP	User Datagram Protocol

Back to Table of Contents

Last Modified: December, 2007
This paper is available at: http://www.cse.wustl.edu/~jain/index.html