Man-in-the-Middle Attack on T-Mobile Wi-Fi Calling

Jethro Beekman
Christopher Thompson

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2013-18
http://www.eecs.berkeley.edu/Pubs/TechRpts/2013/EECS-2013-18.html

March 19, 2013

Man-in-the-Middle Attack on T-Mobile Wi-Fi Calling

Jethro Beekman and Christopher Thompson
Electrical Engineering and Computer Sciences
University of California, Berkeley
jbeekman@eecs.berkeley.edu, cthompson@cs.berkeley.edu

Abstract

T-Mobile has a service called "Wi-Fi Calling", which lets users make and receive calls even when without cellular service. This service is pre-installed on millions of T-Mobile Android smartphones. We analyze the security aspects of this service from a network perspective, and demonstrate a man-in-the-middle attack caused by a lack of TLS certificate validation, allowing an attacker to eavesdrop and even modify calls and text messages placed using the Wi-Fi Calling feature. We have worked with T-Mobile to fix this issue, and, as of 18 March 2013, they report that all affected customers have received an update fixing this vulnerability.

I. Experimental Setup

In order to analyze T-Mobile's Wi-Fi calling system, we used the setup shown in Figure 1. A Wi-Fi calling enabled phone P is configured to use access point AP. AP does not have any wired connections and just acts as a wireless network switch. Another machine M connected to the same network is configured as a DHCP server and NAT router. This allows us record and control all Internet traffic to and from P.

We captured several Wi-Fi calling sessions. During our experiments, all traffic on both network interfaces was captured using libpcap. Any TLS connections were intercepted by sslsniff [1] running on M. sslsniff hijacks a TLS connection request, connects to the remote endpoint itself and generates a certificate based on what it receives from that endpoint. The certificate is signed with any certificate we specify and subsequently sent back to the client. We modified sslsniff to output the master-secret and client-random parameters of all established connections so that the TLS traffic in our packet traces could be decrypted. This allowed us to see how TLS-encrypted messages relate chronologically to other packets.

II. Network analysis

When first enabling Wi-Fi, the DNS conversation in Appendix A takes place, requesting a chain of information about *wifi.msg.pc.t-mobile.com*. Then, a TLS connection is established to the host and port returned by DNS. These are *sba.sipgeo.t-mobile.com* and *5061*, which is the port number assigned by IANA for SIP-TLS.

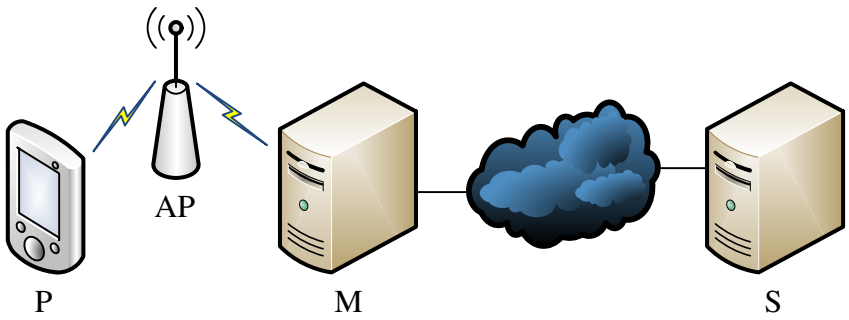

Fig. 1. Wireless man-in-the-middle setup. (P) Phone. (AP) Access point. (M) Man-in-the-middle. (S) Service provider.

1) s:/C=US/ST=WA/L=Bellevue/O=Engineering/CN=*IP*
   i:/C=US/O=T-Mobile USA, Inc./CN=T-Mobile USA, Inc. IP Access Nano 3G CA
2) s:/C=US/O=T-Mobile USA, Inc./CN=T-Mobile USA, Inc. IP Access Nano 3G CA
   i:/C=US/O=T-Mobile USA, Inc./CN=T-Mobile USA, Inc. Root CA
3) s:/C=US/O=T-Mobile USA, Inc./CN=T-Mobile USA, Inc. Root CA
   i:/C=US/O=T-Mobile USA, Inc./CN=T-Mobile USA, Inc. Root CA

Fig. 2. Certificate chain of *sba.sipgeo.t-mobile.com:5061*.

The certificate chain returned by T-Mobile's server is shown in Figure 2. Two things stand out: first, the common name of the first certificate is simply the IP address of the server; second, the self-signed root certificate is not included in standard Certificate Authority (CA) distributions. In fact, searching the web for the exact common name of the root yielded barely any results. This can meant that the root certificate was either built-in to T-Mobile's client software, or they did not implement certificate validation correctly. In fact, the client does not seem to have any problems with sslsniff intercepting the connection, making us conclude the latter. Analysis of the binaries confirmed that there was no trace of the root certificate.

As hinted at by the DNS records and the port number, a SIP [2] dialog is initiated when the TLS connection is established. The client identifies itself using its phone number, IMEI (International Mobile station Equipment Identity) and IMSI (International Mobile Subscriber Identity); see Appendix B for the full message. Further messages follow normal SIP behavior. For example, an INVITE message including the SDP [3] body is outlined in Appendix C, which includes the encryption key that will be used for the SRTP [4] connection. Using the decrypted SIP dialog, an attacker is now able to record all incoming and outgoing calls and text messages (collectively "SIP traffic"). He could record, block and reroute SIP traffic. The attacker could change it by faking a sender or changing the real-time voice data or message content. He could fake incoming traffic and he can impersonate the client with forged outgoing traffic.

We verified the ability to record outgoing calls and incoming and outgoing text messages. We also verified the ability to change the destination phone number on outgoing calls by modifying sslsniff to change all occurrences of <sip:*dest-phone#*@msg.pc.t-mobile.com>, replacing a single target phone number by a different one.

III. Software analysis

T-Mobile has open-sourced most of its Android IMS stack [5]. By exploring the source code, we found two vulnerable components in categories that Fahl et al. [6] have previously identified. *TlsSocketFactory* creates a TLS socket and assigns the *TestTrustManager* to check the validity of certificates. Figure 3 shows a shortened version of the function that verifies the certificates. Note that it does not use an API to extract the common name from the distinguished name and that the only check is whether the common name is in a set of allowed IP addresses. This set was previously initialized with the IP address of the remote endpoint.

```
public void checkServerTrusted(X509Certificate[] chain)
    throws CertificateException
{
    String name = chain[0].getSubjectDN().getName();
    String address = name.replaceFirst("CN=", "").replaceFirst(",.*", "");
    if (!getAllowedRemoteAddresses().contains(address))
        throw new CertificateException("Wrong certificate subject");
}
```

Fig. 3. Implementation of *TestTrustManager::checkServerTrusted*, which checks the validity of a TLS certificate.

IV. Scope

In order to execute man-in-the-middle attacks, an attacker must be on the path of the target network traffic. If they are not already on the path, there are techniques for influencing the path to include the attacker. With ARP spoofing [7] the attacker tricks hosts into believing the attacker's network interface has the router's IP address. Obviously, the attacker can only use this if they are on an on-path network. If the attacker is not, they can use DNS cache poisoning [8], in which they trick a caching DNS resolver to cache an invalid address record for the service they want to attack. Other clients using this same resolver will now receive the malicious record when connecting to this service and will connect to the attacker instead.

Man-in-the-middle attacks would not be as threatening if not for the proliferation of wireless technologies such as Wireless LAN (802.11)—exactly the technology that Wi-Fi calling advertises in its name. In wireless networks, an attacker no longer needs physical access to invade a network. If an attacker can connect to the network they can try ARP spoofing. Or, if the attacker knows the network parameters, they can employ the 'Evil Twin' attack, in which they imitate a legitimate network and trick users into connecting to that network instead. Dai Zovi and Macaulay [9] demonstrated an extended version of this attack exploiting bugs in automatic network selection algorithms in common operating systems.

Not all versions of T-Mobile Wi-Fi calling are necessarily vulnerable. According to T-Mobile's website, the IMS stack is used on the Samsung Galaxy S II, HTC Amaze 4G, myTouch and myTouch Q. The authors have tested the attack on a Samsung Galaxy S Relay 4G and a Samsung Galaxy Note 2. It is likely that other modern T-Mobile Samsung Galaxy products are also vulnerable. Users of T-Mobile Wi-Fi calling for Business might not be vulnerable, since it uses GAN, not IMS technology [10].

V. Patch and Updates

In December 2012, we notified T-Mobile of this vulnerability. Over the past months, they added proper certificate validation to the T-Mobile Wi-Fi Calling feature, so that it validates the identity of the remote endpoint using their self-signed root CA. As of 18 March 2013, T-Mobile reports that they have been able to push an update with this patch to all affected customers. We have independently verified that the update pushed to T-Mobile Android phones successfully prevents this attack.

References

[1] M. Marlinspike, "sslsniff." [Online]. Available: http://www.thoughtcrime.org/software/sslsniff/

[2] "SIP: Session Initiation Protocol," IETF RFC 3261, June 2002. [Online]. Available: https://tools.ietf.org/html/rfc3261

[3] "SDP: Session Description Protocol," IETF RFC 4566, July 2006. [Online]. Available: https://tools.ietf.org/html/rfc4566

[4] "The Secure Real-time Transport Protocol (SRTP)," IETF RFC 3711, March 2004. [Online]. Available: https://tools.ietf.org/html/rfc3711

[5] T-Mobile USA, Inc., "The IMS project for android." [Online]. Available: https://code.google.com/p/the-ims-open-source-project-for-android/

[6] S. Fahl, M. Harbach, T. Muders, L. Baumgärtner, B. Freisleben, and M. Smith, "Why Eve and Mallory love Android: an analysis of Android SSL (in)security," in *Proceedings of the ACM conference on Computer and communications security*, 2012, pp. 50–61.

[7] R. Siles. (2003) Real World ARP Spoofing. SANS. [Online]. Available: http://pen-testing.sans.org/resources/papers/gcih/real-world-arp-spoofing-105411

[8] S. M. Bellovin, "Using the domain name system for system break-ins," in *Proceedings of the 5th conference on USENIX UNIX Security Symposium*, 1995.

[9] D. A. Dai Zovi and S. A. Macaulay, "Attacking automatic wireless network selection," in *Proceedings from the 6th Annual IEEE SMC Information Assurance Workshop*, June 2005, pp. 365–372.

[10] Cisco Systems, Inc. (2010) T-Mobile Wi-Fi Calling for Business with Cisco Unified Wireless Network. Cisco Partner Solution Profile. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns340/ns394/ns348/ns828/c22-638810-00_tMob_sBrief.pdf

Appendix

A. Initial DNS conversation from Wi-Fi calling client

```
> NAPTR wifi.msg.pc.t-mobile.com
< NAPTR 100 10 S SIPS+D2T _sips._tcp.sba.sip.t-mobile.com
```

```
> SRV _sips._tcp.sba.sip.t-mobile.com
< SRV 10 10 5061 sba.sipgeo.t-mobile.com

> A sba.sipgeo.t-mobile.com
< A <IP>

> PTR <reverse(IP)>.in-addr.arpa
< PTR m<hex(IP)>.tmodns.net
```

B. Initial REGISTER message from Wi-Fi calling client

```
REGISTER sip:msg.pc.t-mobile.com SIP/2.0
To: <sip:src-phone#@msg.pc.t-mobile.com>
Max-Forwards: 70
Supported: 100rel, eventlist
User-Agent: T-mobile TAS
P-Last-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=ID#1
P-Access-Network-Info: IEEE-802.11; i-wlan-node-id=ID#2
P-Preferred-Identity: sip:src-phone#@msg.pc.t-mobile.com
Call-ID: UUID#1@src-IP
From: <sip:src-phone#@msg.pc.t-mobile.com>;tag=UUID#2
CSeq: 1 REGISTER
Content-Length: 0
Via: SIP/2.0/TLS src-IP:src-port;branch=UUID#3
Contact: <sip:src-phone#@src-IP:src-port;transport=TLS>;expires=3600;
    reg-id=1;+sip.instance="<urn:gsma:imei:IMEI;svn=00>";
    +g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.mmtel";
    +g.3gpp.smsip;
    +g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.e-location"
Authorization: Digest username="IMSI@msg.pc.t-mobile.com",
    realm="msg.pc.t-mobile.com", nonce="",
    uri="sip:msg.pc.t-mobile.com", response="", algorithm=AKAv1-MD5
Privacy: none
```

C. INVITE message for call originating from Wi-Fi calling client

```
INVITE sip:dest-phone#@msg.pc.t-mobile.com;user=phone SIP/2.0
To: <sip:dest-phone#@msg.pc.t-mobile.com>
Max-Forwards: 70
Supported: 100rel
User-Agent: T-mobile TAS
P-Early-Media: supported
P-Last-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=ID#1
P-Access-Network-Info: IEEE-802.11; i-wlan-node-id=ID#2
Call-ID: UUID#1@src-IP
From: <sip:src-phone#@msg.pc.t-mobile.com>;tag=UUID#2
CSeq: 1 INVITE
Content-Length: size
Via: SIP/2.0/TLS src-IP:src-port;branch=UUID#3
Contact: <sip:src-phone#@src-IP:src-port;transport=TLS>;
    +g.3gpp.icsi-ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.mmtel";
```

```
    +sip.instance="<urn:gsma:imei:IMEI;svn=00>"
Allow: INVITE, CANCEL, ACK, OPTIONS, BYE, PRACK, UPDATE, NOTIFY,
    REFER, MESSAGE
Accept-Contact: *;
    +g.3gpp.icsi_ref="urn%3Aurn-7%3A3gpp-service.ims.icsi.mmtel";
    +g.3gpp.smsip
Content-Type: application/sdp

v=0
o=- session-ID 1 IN IP4 src-IP
s=-
t=0 0
m=audio src-port RTP/SAVP 0
i=StreamMediaAudio
c=IN IP4 src-IP
a=crypto:1 AES_CM_128_HMAC_SHA1_32 inline:base64-encoded key (16B) and salt (14B)
a=rtpmap:0 PCMU/8000
a=sendrecv
```