

Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 24, 2016

W. Kumari
Google
W. George
Time Warner Cable
February 21, 2016

OWE: Opportunistic Wireless Encryption
draft-wkumari-owe-02

Abstract

This document describes a method to incrementally increase the security of wireless networks against passive attackers / pervasive monitors through unauthenticated encryption.

[Ed note: Text inside square brackets ([]) is additional background information, answers to frequently asked questions, general musings, etc. They will be removed before publication.]

[This document is being collaborated on in Github at: <https://github.com/wkumari/draft-wkumari-owe>. The most recent version of the document, open issues, etc should all be available here. The authors (gratefully) accept pull requests.]

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 24, 2016.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

Internet-Draft

[draft-wkumari-owe](#)

February 2016

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	tl;dr / Executive summary	2
1.1.	FAQ / Common questions / Notes	3
2.	Introduction / Background	4
2.1.	Requirements notation	5
3.	OWE protected networks	5
3.1.	OWE Support Advertisement in Beacons	5
3.2.	OWE Advertisement in Access Network Query Protocol (ANQP)	6
4.	Deployment	6
4.1.	Advertising OWE support on legacy devices	7
5.	Implementation notes	7
6.	IANA Considerations	8
7.	Security Considerations	8
8.	Privacy Considerations	9
9.	Acknowledgements	9
10.	References	9
10.1.	Normative References	9
10.2.	Informative References	9
Appendix A.	Changes / Author Notes.	10
	Authors' Addresses	10

[1.](#) [tl;dr / Executive summary](#)

This (colloquial) summary will be removed before publication:

Currently, there are many open/unencrypted public WiFi networks, designed for "ease of use" in places such as coffee shops, libraries, etc. When a user connects to these open networks, they are using an unencrypted connection and their traffic can be easily viewed and/or intercepted by any other user within wireless range, simply by using a network sniffing tool such as Wireshark.

While users *should* use a VPN, many do not. Places that provide public WiFi access *should* only provide an encrypted SSID, and print the passphrase on the wall or receipts (or similar); but, well, they are a coffee shop, and want to provide easy access to everyone who buys an espresso...

This document defines a new, opportunistically encrypted mode for WiFi networks. The SSID will still appear to be open (there will not be a "lock" icon when scanning), but will actually be encrypted, using the SSID as the passphrase. Software running on the client will detect that this is an opportunistically encrypted connection and will automatically provide the SSID as the passphrase, providing an encrypted connection without any interaction from the user. This system leverages existing WiFi protocols and WPA2, the only change is in operational practices and the client UI.

[1.1.](#) FAQ / Common questions / Notes

Q1: If everyone uses the SSID as the key, can't attackers just use this to decrypt all the data?

A: WPA2 using PSK generates a unique Pairwise Transient Key (PTK) between the AP and each client. This PTK is derived using something called the 4-Way Handshake (see IEEE 802.11-2012 (or the summary at https://en.wikipedia.org/wiki/IEEE_802.11i-2004#The_four-way_handshake)), which includes nonces from both the client and the AP. Unfortunately, this doesn't use a public key exchange protocol, and so an attacker who can watch the initial client association can derive the user's encryption key. This is a weakness in WPA2-PSK, and not specific to OWE.

Q2: So does this actually help?

A: Yes. This provides protection if the passive attacker wasn't already present when the user connected, or if the passive attacker was not able to hear both sides of the connection. OWE does not provide very strong protection, and does not claim to. It does, however, raise the bar for the attacker, or force him to become active (and force users to disassociate (so he can watch the 4 way handshake)). OWE specifically does NOT provide the "lock" icon (or any other obvious feedback) when users scan for open wireless networks, because we do not want users to assume that they are getting "real" encryption. OWE will become more secure if and when WiFi with a public key exchange (such as Diffie-Hellman) is deployed.

The incremental cost to implement OWE is very small (it involves adding a flag to beacons, and clients to know to not display the lock icon) and so we feel that the benefit outweighs the cost.

Q3: Isn't this vulnerable to the fake AP attack?!

A: Yes. An attacker can stand up their own AP with the same SSID and passphrase. This is true for any network that uses WPA2-PSK when the attacker knows the passphrase (for example, if the coffee shop prints the passphrase on the wall, receipts, etc) and it not specific to OWE. OWE is not designed to defeat active attackers, nor solve all issues. See Q2.

Q4: This isn't really opportunistic encryption...

A: Perhaps not, but it is has many of the same properties - it is unauthenticated, is mainly designed to deal with passive listeners, doesn't require interaction from the user, etc. I also wanted to have a cool acronym - actually I wanted it to be OWL, but was not able to reverse engineer a non-contrived name that made that...

Q5: Doesn't this belong in [IEEE | WiFi Alliance | <insert other SDO here>] ?

A: Answer unclear, ask again later. I have discussed this with a number of people who participate in other SDOs, and it seems like the IETF is the best home for it, at least for now. It does not require changes to any underlying transport, it does not change any standards, it simply takes advantage of work done in other standards bodies.

[2.](#) Introduction / Background

As of the time of this writing, it is very common for users to connect to so called "open" wireless networks, for example in coffee shops, hotels, airports and similar. These networks provide no encryption, which means that the user's traffic is visible to anyone nearby with packet capture software, for example, Wireshark. It is also trivial for an attacker to perform a Man-in-the-Middle attacks as they can see all of the user's traffic.

There are a number of solutions to this problem, such as WPA2 (Wi-Fi Protected Access II), but these require either obtaining a passphrase from the network operator (WPA-Personal / Pre-Shared Key (PSK) mode)

or having valid credentials for the network (WPA-Enterprise / [\[IEEE.802-11i\]](#)WPA-802.1X).

While these provide good security, for convenience reasons network operators often deploy open / unencrypted networks for public or "guest" use. This allows the public or visitors to get Internet access without having to ask for a passphrase, look around for one printed on a receipt or similar. Instead of chastising network operators for providing insecure access, this document provides a method to implement unauthenticated, encrypted network access.

This is not intended to replace other existing and more robust methods of authentication that provide encrypted access to a WiFi network once the user is authenticated and authorized to join the network, e.g. WPA-Enterprise / IEEE 802.1x or Hotspot 2.0. Rather, this is intended for low-end, unmanaged guest access networks such as SOHO networks that would otherwise either be left unencrypted, or whose password would be shared via other means such as posting it on the wall of the coffee shop.

[2.1.](#) Requirements notation

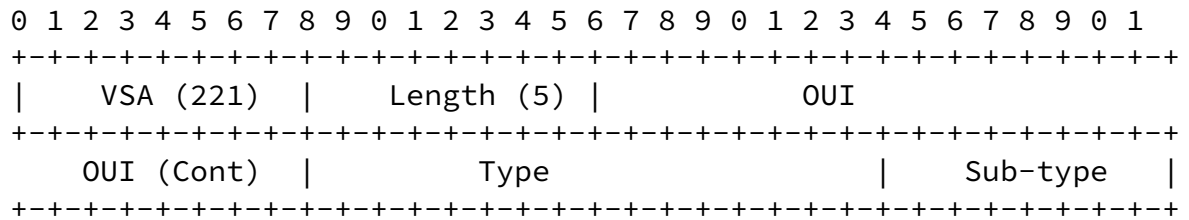
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [\[RFC2119\]](#).

[3.](#) OWE protected networks

In order to provide an encrypted connection using OWE, the network operator creates an SSID with the (WPA2 / 802.11i [\[IEEE.802-11i\]](#)) pre-shared key identical to the SSID. As part of the WPA2 protocol the wireless client (STation) and Access Point (AP) derive a Pairwise Transient Key (PTK), which provides an encrypted "channel" between the client and AP.

[3.1.](#) OWE Support Advertisement in Beacons

In order to advertise that this network supports OWE the Access Point will include the OWE Vendor-specific Information Element in Wireless Beacon frames.



VSA One octet. The IEEE Assigned Vendor-Specific Information element ID - 221.

Length One octet. The length of the information field, including the OUI - 5.

OUI Three octets. The OUI for the of the entity that has defined the content of the particular vendor-specific information element - 64-6A-74 (AUTH-SERVERS). [If approved, this may move under the IANA OUI if desired]

Type One octet. OWE has been assigned Type 1 under the AUTH-SERVERS OUI

Sub-type One octet. Sub-type identifies the version of the OWE protocol. Currently only 0 is defined.

An Access Point that includes the OWE Vendor-specific Information Element in beacon frames is signalling that it supports OWE on that particular SSID, and that the PSK is the same as the SSID. Client (STations) connecting to OWE enabled networks MUST use the SSID as the PSK, and MUST NOT display a "lock" icon in the list of SSIDs when scanning. User Interfaces MAY provide some feedback that this is an OWE protected network, but this should not be too prominent to avoid users assuming that they are getting more security than they actually are.

[3.2.](#) OWE Advertisement in Access Network Query Protocol (ANQP)

This section to be fleshed out later, but the same general principle applies.

Support for OWE can also be advertised in IEEE 802.11u-2011 using a virtual roaming consortium with the same OUI. Examples will be provided soon.

[4.](#) Deployment

Because one of the largest problems with most low-end WiFi devices is their ability to receive timely updates to patch security holes and add new features post sale, it may be appropriate to define a lightweight version of this opportunistic encryption such that one or both sides of the wireless network connection can take advantage of this improved privacy via opportunistic encryption despite not being updated to formally support OWE beacons. This model simply defines an agreed-upon or best practice method for manually configuring both network and client devices to attempt connecting to open, but secured WiFi networks when the password is not published, but the presence of a password is intended to provide link encryption rather than access control.

As with the full mode defined above, the access point is configured to accept a WPA2-PSK that is identical to the SSID. However, instead of advertising the OWE capability in beacons, the network looks like a standard encrypted network to host devices that wish to connect to it. Host devices that are not OWE aware can be configured by the user to connect in the standard process, by selecting the desired SSID and manually entering a password that just happens to be the same as the SSID. Host devices that are OWE-aware can automatically try the SSID as a password when the user selects that network to attempt to connect to it, and only present the user with a password prompt if that authentication fails, even if there is no OWE beacon seen from the AP. If the device is able to connect to the network automatically via the SSID password, it can infer that this is an OWE network and present the appropriate notifications to the user.

[4.1.](#) Advertising OWE support on legacy devices

OWE is designed for use on consumer / commodity Customer Premises Equipment (CPE). The software running on these devices are often not upgraded for many years, and so adding the OWE Vendor Specific Information Element into the beacon may not be feasible. In order to allow users of these devices to still be able to advertise OWE support we define an interim measure.

A user who wishes to advertise that their network / SSID supports OWE should add the string '_owe' to the SSID name and set the passphrase the same as the SSID. Clients connecting to this SSID SHOULD try the full SSID name (including the _owe suffix) and SHOULD NOT display the lock icon. Users who have legacy clients can manually see that the SSID ends in _owe and manually configure the passphrase as the SSID.

5. Implementation notes

[Ed note: This section contains rough notes for people who want to experiment with OWE. It will be tidied / removed before publication.]

There is some (very rough) example code in the Github repository, and also some example beacon captures, in pcapng format (view with Wireshark / tcpdump)

The easiest way to quickly test this (IMO) is to install the hostapd tools on something like a Raspberry Pi, and then add

```
<CODE BEGINS>
#OWE:
vendor_elements=dd05646a740100
<CODE ENDS>
```

to /etc/hostapd/hostapd.conf.

Another easy option is to use an AP running OpenWRT[OpenWrt]. My testing setup for this is a Ubiquiti Unifi (~\$70USD on Amazon) running Barrier Breaker 14.07.

After installing OpenWRT login via SSH and edit the /lib/netifd/hostapd.sh (this gets run when the WiFi interfaces is enabled). Find the section around 'append bss_conf' and add:


```
#This adds the OWE 802.11 Vendor Specific Information Element to the beacon frame
append bss_conf "# OWE: Opportunistic Wireless Encryption - draft-wkumari-owe"
append bss_conf "vendor_elements=dd05646a740100" "$N"
<CODE ENDS>
```

Disable and reenable the Wireless interface and it should start including the OWE information element in all beacon frames. You can look at the generated config in /tmp/run/hostapd-phy0.conf. While it works, this code is far from ideal - it always includes the OWE Vendor Specific Information Element - eventually I'll add something to the GUI to enable users to toggle it on and off, but this is a good start for testing. Look for additional code in the Github repo soon!

6. IANA Considerations

[To be completed after discussions]

Currently the OWE Vendor-specific Information Element is using type 1, sub-type 0 under the AUTH-SERVERS OUI. This is to allow experimentation with OWE without squatting on the IANA OUI. If OWE progresses within the IETF, and the IESG chooses, I'm fine to place this under the IANA OUI, or for it to remain under AUTH-SERVERS. It's all just numbers.

7. Security Considerations

There are many attacks that this does not protect against, including attackers watching the 4-Way Handshake and deriving the PTK between the client and the user. This is a weakness in the wireless specification, and not specific to OWE. In order to not have the user assume that they are getting stronger protection than they really are, the user interface should not provide obvious feedback that OWE is in use. OWE simply raises the bar slightly; it does not claim to solve all wireless issues.

This solution does not protect against so called "fake AP" attacks. Wireless networks that use PSKs that the attacker may know are vulnerable to an attacker standing up an access point with the same SSID and PSK. This is not specific to OWE, it affects all WiFi networks.

This solution does not (directly) protect against disassociation attacks and the attacker observing the client authentication. This is not specific to OWE.

This solution does not claim to provide "strong" security, it is intended to be less insecure than "open" WiFi. In order to avoid users assuming that they are getting more security than they really are, OWE protected networks do not get a "lock" icon then scanning for WiFi networks.

Ideally users would only associate to networks that they trust, using WPA2-Enterprise (802.1X) with certificates that they trust, and then immediately use a VPN to a trusted endpoint. However, open wifi is really convenient and users will continue to want it. While abstinence is the best policy, OWE recognises that users will continue to behave in risky ways, and thus aims to make this slightly less risky...

[8.](#) Privacy Considerations

By making "open" wireless encrypted by default we aim to increase privacy by decreasing the incidence of passive eavesdropping by pervasive monitors and idle attackers.

[9.](#) Acknowledgements

The authors would like to thank a bunch of people, including Stephen Farrell, Ted Hardie, Chris Morrow.

[10.](#) References

[10.1.](#) Normative References

- [IEEE.802-11i] IANA, "IEEE 802 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 6: Medium Access Control (MAC) Security Enhancements", <<http://standards.ieee.org/getieee802/download/802.11i-2004.pdf>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.

[10.2.](#) Informative References

- [OpenWrt] IANA, "OpenWrt", <<http://wiki.openwrt.org/start>>.

Internet-Draft

[draft-wkumari-owe](#)

February 2016

[Appendix A](#). Changes / Author Notes.

[RFC Editor: Please remove this section before publication]

From -01 to -02:

- o Integrated some changes from Wes.

From -00 to -01:

- o Included comments and feedback from Stephen Farrell.
- o Some more nits.

From null to -00.

- o Initial text.

Authors' Addresses

Warren Kumari
Google
1600 Amphitheatre Parkway
Mountain View, CA 94043
US

Email: warren@kumari.net

Wesley George
Time Warner Cable
13820 Sunrise Valley Drive
Herndon, VA 20171
US

Email: wesley.george@twcable.com

