# BadUSB 2.0: Exploring USB Man-In-The-Middle Attacks[1]


## Authors:

**David Kierznowski**, MSc (Royal Holloway, 2015)
**Keith Mayes**, ISG, Royal Holloway



# Abstract

The uses and capabilities of rogue USB hardware implants for use in cyber espionage activities is still very much an unknown quantity in the industry. Security professionals would benefit from tools capable of exploring the threat landscape while increasing awareness and countermeasures in this area. BadUSB 2.0 or BadUSB2 is such an investigative tool capable of compromising USB fixed-line communications through an active Man-In-The-Middle attack. It is able to achieve the same results as hardware keyloggers, keyboard emulation devices and earlier BadUSB hardware implants, thus providing an insight into how these attacks may be prevented. Furthermore, BadUSB2 is able to evaluate new techniques to defeat keyboard-based one-time-password systems, automatically replay user credentials, as well as acquire an interactive command shell over USB.


# Introduction

Embedded devices are being targeted either through rogue hardware implants or placing malicious code directly into the firmware. Stuxnet was a wakeup call that USB and other embedded devices can be used to compromise any system including air-gapped networks. The Internet of Things (IOT) and vehicular security with remotely controlled cars will only increase interest in this space.

The evaluation tool, BadUSB2, was developed as a means to evaluate the compromise of USB fixed-line communications through an active Man-In-The-Middle (MITM) attack. Unlike the BadUSB attack released in 2014, which targets USB firmware, BadUSB2 targets the USB cable. With the use of two bespoke USB hardware devices costing around $100 each, it is possible to perform active MITM attacks to eavesdrop, modify, replay and fabricate messages between a USB keyboard and host. Furthermore, from an operating-system perspective, only the “legitimate” keyboard is visible making it

1 This article is to be published online by Computer Weekly as part of the 2016 Royal Holloway information security thesis series. It is based on an MSc dissertation written as part of the MSc in Information Security at the ISG, Royal Holloway, University of London. The full MSc thesis is published on the ISG's website.

difficult to detect and block access without physical inspection.

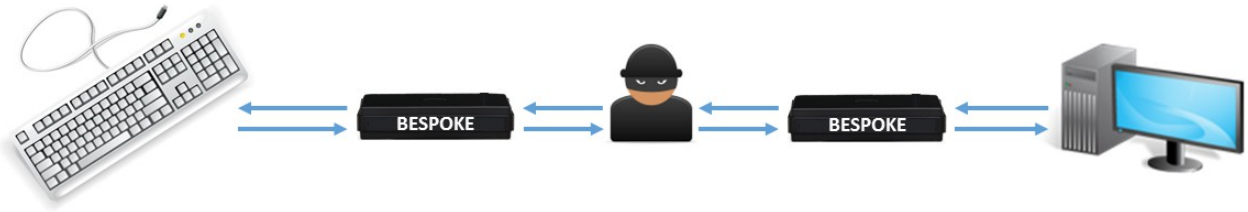



*Figure 1- The BadUSB2 Architecture*[2]

The improvements in network access controls and endpoint security systems limit the attack surface available to traditional attacks involving rogue wireless access points or more recently the keyboard emulating devices like "Rubber Ducky". BadUSB and now BadUSB2 demonstrate a new breed of USB based attacks that not only attempt to circumvent existing controls, but also allow an adversary to access the network with user privileges. In addition, BadUSB2 can mimic the role of other hardware implants, namely, hardware keyloggers, keyboard emulating devices or even BadUSB. Furthermore, it can combine these attacks in real-time which allows analysis of several new attacks.

# Attacks Demonstrated with BadUSB2

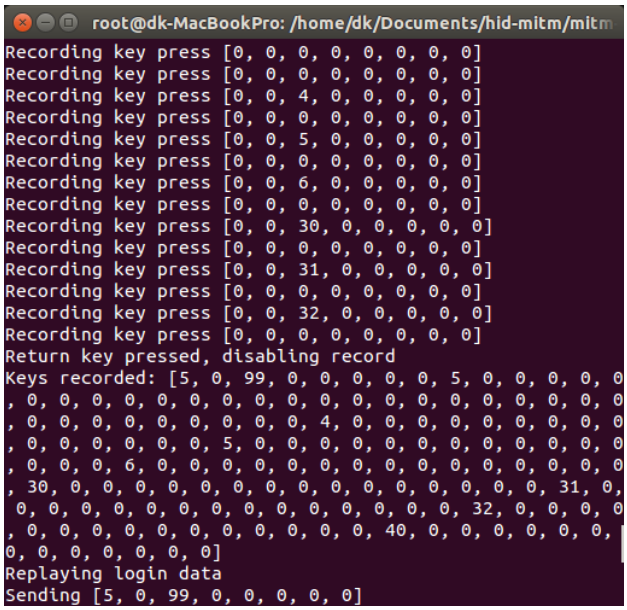
The author has been able to practically demonstrate each of the attacks described below within a lab environment (with the exception of BadUSB which was considered out of scope). However, the reader should be aware that similar functionality used by a real attacker could incorporate over the air technology as seen in the NSA's COTTONMOUTH-I keyboard implant.

**Eavesdropping.** The most obvious attack when observing user keystrokes is to simply record them. This would allow the adversary to recover login credentials and any confidential data typed out by the user.

**Sending Keystrokes.** As the device targeted is a keyboard, the adversary can send commands directly to the operating-system, emulating a user typing out a message. A large number of malicious scripts are already available for test with BadUSB2 due to the "Rubber Ducky" project. It is common for the adversary to use this attack to spawn a new shell over the network.

---

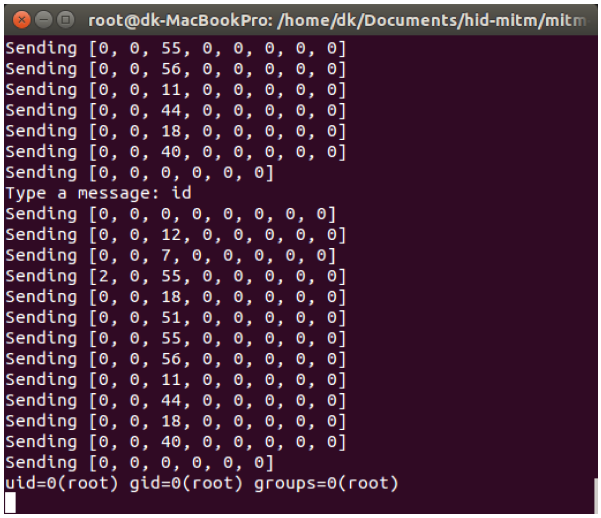
2 The figures in Figure 1 are taken from:
Bespoke (router): http://www.dlink.com/uk/en/home-solutions/connect/routers/dir-600-wireless-n-150-home-router
Workstation: http://www.iconarchive.com/show/network-icons-by-devcom/workstation-Vista-icon.html
Keyboard: http://all-free-download.com/free-vector/download/white_computer_keyboard_vector_147845.html
https://www.globalsign.com/en/blog/lenovo-enables-main-in-the-middle-attacks-via-superfish-adware/




**Replaying Logon Credentials.** With the ability to capture, record and replay user keystrokes, an adversary could record the user logging in and then replay these keystrokes at a later time. Logins that use two-factor authentication would be immune to this attack. However, the adversary could still observe the user's actions or attempt to send a malicious action after the user has authenticated.

**Character Substitution Attack.** BadUSB2 can demonstrate the effect of an adversary modifying messages sent between the USB keyboard and the host in real-time. For example, a Character Substitution Attack would allow an adversary to defeat keyboard-based One-Time Password (OTP) systems by capturing the legitimate OTP while modifying the user's version of the OTP, causing it to fail.

**Data Exfiltration (Covert Channel).** It has been known for some time that data can be exfiltrated over USB-HID (Human Interface Device), the protocol used by keyboards and other types of USB devices. This allows an adversary to capture and copy screenshots or other files. This provides a unique covert channel as the adversary would not need to exfiltrate data over the network.




**Interactive Shell over USB-HID (Covert Channel).** Using the data exfiltration techniques already discussed, BadUSB2 can be used to acquire an interactive shell over USB-HID. The adversary sends a command and simply retrieves the output using the same techniques used in the data exfiltration attack.

**BadUSB.** As the adversary is physically connected to the host's USB port it would be possible to disconnect as a keyboard and attach to the operating-system as a different device achieving the same results as BadUSB.

# Countermeasures to Attacks Demonstrated by BADUSB2

The primary defence techniques used against rogue USB hardware implants by enterprise endpoint security systems include:

- Whitelisting to ensure that only permitted devices can be connected;
- Secondary Device detection that checks when more than one type of device has been attached, for example two keyboards or two network cards.

Used together, these countermeasures can mitigate the threats of keyboard emulation attacks and BadUSB. Security systems would be able to detect and block when an additional (secondary) USB device is attached. However, an adversary device with similar capabilities to BadUSB2 would not be affected by these countermeasures. BadUSB2 targets USB fixed line communications, so no “new” secondary devices are connected.

Although it is not possible to completely mitigate BadUSB2 attacks without a cryptographic solution, focusing on securing people, processes and technology could go a long way to reduce the overall risk. The following recommendations should be considered:

- **Antivirus/Application Whitelisting.** In order to acquire an interactive shell over USB the adversary must first copy ("type out") an executable to the host operating-system to perform the data exfiltration. This code is then visible and could be flagged up by antivirus software or restricted through application whitelisting.
- **Heuristics.** The following detection techniques could be used:
    - In order to exfiltrate data, the USB protocol will be used in a non-standard way. A threshold on the number of caps-lock, num-lock and scroll-lock keys could be sufficient to trigger an alert.
    - USB devices can perform several functions. Each function has its own communication endpoint number and is hardcoded into the firmware. The bespoke hardware devices used to perform BadUSB2 attacks also use hardcoded endpoints. If these endpoint numbers are not the same on both the USB peripheral and bespoke hardware the adversary will be required to modify the endpoint numbers during the USB setup phase. If a legitimate USB peripheral suddenly changes endpoint numbers there is a strong likelihood that an attack has just started. This technique can also be used to forensically determine when an exploit begins.
    - Identifying changes in voltage or the time taken for USB enumeration may also be used as early warning attack indicators.
- **Two-factor authentication.** If the user is required to present something the user has during the login process the adversary device would not be able to simply replay the user's credentials.
- **Physical Security.** The adversary would need to be in a position to setup the hardware in order to achieve this attack. Therefore physical security including looking at supply-chain vulnerabilities could mitigate these threats. In addition, simply using laptops instead of desktops may make this attack impractical, unless the user attaches an external USB keyboard or docks the laptop.
- **User Awareness.** Users should be trained to regularly check their computers for obvious types of hardware implants.
- **Rogue Device Checking.** Technical audits for rogue wireless access points could include spot checks for rogue hardware implants through physical inspection.
- **Cryptography.** The attacks investigated via BadUSB2 are only possible due to a lack of

firmware code signing and data origin authentication. Applying an appropriate cryptographic protocol could counter these attacks. However, due to cost and complexity it may be some time before we see such solutions implemented across all USB devices.

# Conclusion

The development and use of the BadUSB2 tool has highlighted the need for countermeasures to powerful known attacks against the USB interface. The tool is sufficiently flexible to investigate emerging threats, which is important as the USB interface/Devices will proliferate, yet remain vulnerable. It is hoped that the tool will help educate and convince equipment vendors to place emphasis on information security best-practices as part of the system and process design and not as an after-thought. Organisations may also be made more aware of USB vulnerabilities and consider defence-in-depth strategies against insider threats as well as against their external boundaries.

## Biographies

*David Kierznowski* has over a decade of experience as an information security professional working in both the private and public sectors in the United Kingdom and the Middle-East. He has fulfilled roles such as principal security consultant, technical team lead and trainer. He holds several professional qualifications and an MSc in information security from Royal Holloway University.

*Keith Mayes* B.Sc. Ph.D. CEng FIET A.Inst.ISP, has spent much of his life working in/with industry, yet is also an active researcher/author with 100+ publications. He is Director of the ISG at Royal Holloway University of London and of Crisp Telecom Limited. He has worked in hardware/software development, DSP and sensors, standardisation, mobile communications, smart cards/RFIDs, embedded systems, systems modelling plus diverse aspects of information security.