27Mhz Wireless Keyboard Analysis Report aka “We know what you typed last summer”

Authors: Max Moser & Philipp Schrödel
(Dreamlab Technologies AG & Remote-exploit.org)

Table of Contents

DREAMLAB
TECHNOLOGIES

27Mhz Wireless Keyboard Analysis Report aka “We know what you typed last summer”

Authors: Max Moser & Philipp Schrödel
(Dreamlab Technologies AG & Remote-exploit.org)

Summary

Wireless keyboards have been distributed for years all over the globe. After the initial infraded based keyboards, the vendors developed radio frequency based models operating at 27Mhz. Logitech and Microsoft are two major brands in this market area. Their products are sold in many consumer electronic stores worldwide.

After of analyzing wireless keyboard communication, Dreamlab is able to understand their functionalities, eavesdrop their traffic, crack the encryption key and decrypt the data into clear text keystrokes. The keystokes from any analyzed keyboard within the radio receiver's range can be sniffed at the same time.

The above statement is true and validated for Microsoft's Wireless Optical Desktop 1000 & Wireless Optical Desktop 2000 products. Unfortunately we could not validate it against all of the Microsoft models but according to the product documentation and pictures available on the internet, the attack might also work on the following modells: Wireless Optical Desktop 3000, Wireless Optical Desktop 4000 as well as their 27Mhz based Wireless Laser Desktop series.

Please note that this document contains information about the named keyboards, other brand/products/models might differ. A detailed analysis of Logitech models is still in progress and will be published when available. We are aware that there is no quick fix for this hardware design vulnerability so we decided **not to release the proof of concept to the public** and **we dont release the full protocol details at the moment,** but maybe after we finish the research on other brands and the new solutions like Logitech's “Secure Connect”.

Radio Frequencies are shared media and should be considered to be shared. We suggest to not use insecure communication channels for important information without adequate levels of encryption.

Dreamlab is willing to demonstrate the attack on request and will publish a demonstrational video on their website. In addition, the researchers have created a presenation about their work, the procedures used and the pitfalls they experienced during the analysis. They will present their work at different events or you can book them for individual educational presentations/trainings. This will hopefully help researchers get into this very interesting topic of analyzing unknown radio based data transmission.

Details

In this part of the document we present some details on the functionalities of the keyboards. Please note that this content is writen based on the interpretation of the data traffic and the behavior. There is still a possibility of misinterpretation or other errors because we never had access to the binaries or the code used in the keyboard solutions.

Important things first, we would like to especially offer thanks to Hunz for his great help and his patience with us ignoramus people :-) as well as toDreamlab for providing us research time and ressources to conduct the analysis.

Historic Publication

To our knowledge, there is no publication freely available regarding the encryption or security features in wireless keyboards. In 2001 http://www.datentreuhand.de did publish a product design fault related to thosed keyboards. The announcemet described the problem that more than one receiver can be associated to a wireless keyboard. This problem still exists in all products we haved analyzed so far, because they still associate using a pure one-way communication model. We guess that the vendors did not react to this publication because a successful attack does depend on the knowledge of the model used by the victim (both receivers need to be the same model) and the evesdropping of the whole keyboard association process (e-association could be enforced using a jammer).

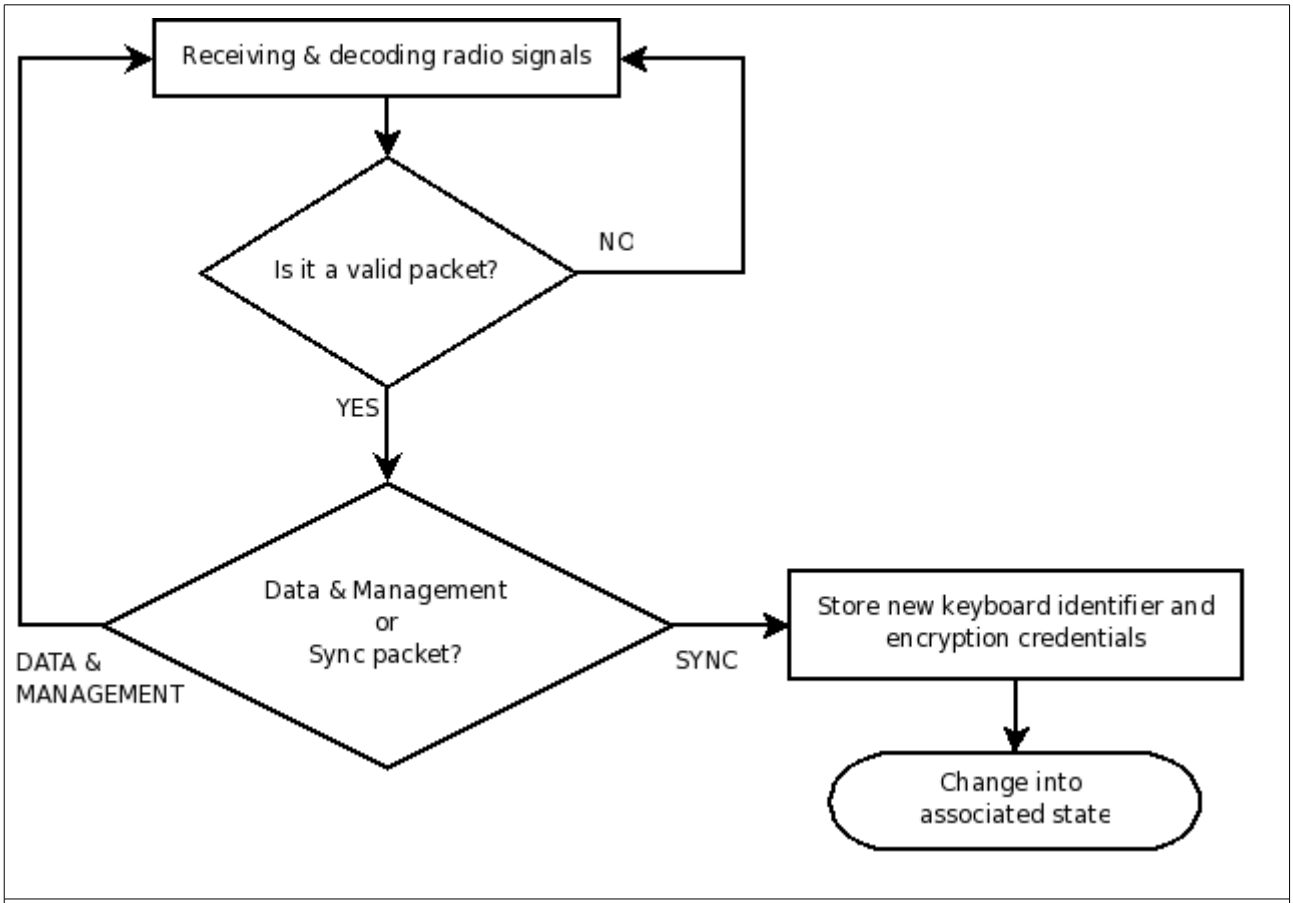
Communication Basics

When a new keyboard is intended to be used on a computer, the corresponding receiver has to be attached to the computer and the association process has to be conducted to ensure complete functionalities. This is usually done by pressing a “connect” or “synch” button on the receiver as well as on the keyboard. The receiver now captures and decrypts the keystrokes from its associated keyboard. Packets origniated by other keyboards within receiving range will be ignored. The behavior of the receiver can be distinguished into two communication states:

- Unassociated (The receiver is not associated with a specific keyboard)
- Associated (The receiver is associated with a specific keyboard)

Unassociated State:

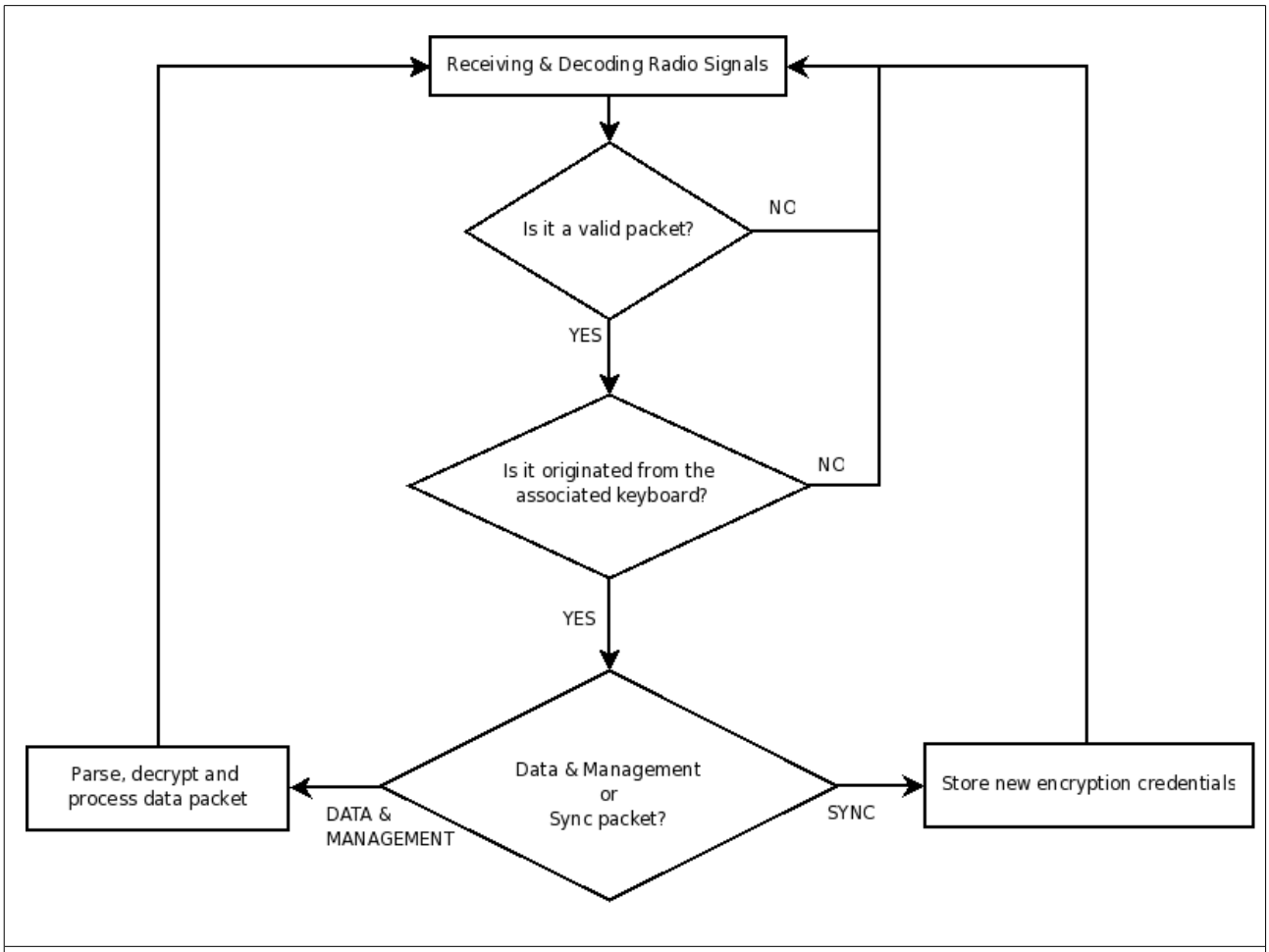
The receiver contains EEPROM to store information about the last keyboard associated to it. By pressing the “connect” button on the receiver, it will get into the unassociated state and accepts “Snychronization packets” from any keyboard within range. The first successfully received synchronization packet will be interpreted and the association data will be saved into the EEPROM and the receiver is switched into associated state.



Picture 1: Simplified Flowchart Of The Receiver's Workflow While In Unassociated State

Associated State:

During this state, the receiver verifies a keyboard identifier in every paket it receives. If the identifier bits do not match, the packet is ignored. This is true for data packets as well as synchronization pakets. When a keyboard is sending synch packets and the relevant associated receiver is receiving them, they are accepted. This will not cause a full re-association but it changes the encryption key used. (Details about ID and encryption will follow)



Picture 1: Simplefy Flowchart Of The Receiver's Workflow While In Associated State

DREAMLAB
TECHNOLOGIES

Protocol Details:

We are not releasing the full details yet, because our analysis is not completely finished at the moment.

Its seems that there are three main paket frametypes, Data Packets (Transfering actual keystroke data), Management Packets (End of transmission frames) and Synch Packets (Associate a receiver with a specific keyboard). Each of the packet types are roughly outlined within idividual sections.

Synch Packets:

The Synch packets are used on at least two occasions:

- Associating a receiver with a specific keyboard by pressing the connect buttons on the receiver **as well as** on the keyboard.
- Change the encryption credentials of an already associated receiver & keyboard pair by pressing the connect button **only on the keyboard**, while the receiver is in normal operating mode.

A Synch packet contains the following information:

- “Universal” static information, which is allways identical among multiple keyboards (Might be a manufaturer identifier).
- Keyboard specific, static information (Might be the keyboard identifier)
- Keyboard specific, random information, that changes during each association/synch procedure (Contains encryption keys)
- Some yet unknown bits (Seems to be some kind of an error detection mechanism)

Data Packets:

The Data packets are used to transport keystroke data from the keyboard to the receiver. The keystroke (USB HID keycode) is transported in encrypted form.
Metakeys like “Shift”, “CTRL” and “ALT” are not encrypted and are transported as flagfields.
When the receiver gets the data packet and successfully validates the originator, it decrpyts the keystroke and sends it to the computer in cleartext.

- Data Packets seem to consist of the following information:
- Keyboard specific, static information (Might be the keyboard identifier)
- The actual encrypted USB HID keycode
- State flags to distinguish a key press and keyrelease of the HID keycode
- A field of flags representing the different metakeys
- Some yet unknown bits (Seems to be some kind of an error detection mechanism)

Management Pakets:

To identify the moment when all keys have been released, the protocol has an “End of transmission” management packet. These packets are identical among all the keyboards except for the originated keyboard identifier. This means that there is no encryption involved at all.

- Originating keyboard identifier (Might also be some kind of a checksum)
- Static EOT packet content (Unknown meaning of content bits)

Encrytption Details:

To our surprise, only the actual keystroke data seems to be encrypted. The Metaflags and identifier bits aren't encrypted or obfuscated.

The one byte USB Hid code is encrypted using **a simlple XOR** mechanism with a **single byte of random data** generated during the association procedure.

This means that there are only 256 different key values possible per keyboard and receiver pair. We did not notice any automated key change interval and therefore assume that the encrpytion key stays the same until the user reassociates the keyboard.

256 key combination can be bruteforced even with very slow computers today. We did not analyze the quality of the random number so far because it was not needed to successfully break the encryption.

Our Proof Of Concept Tool

During our development of the proof of concept tool, we evaluated different methods to reduce the number of possible valid keys. After a short testing time, we used the simplest but most successful procedure. Using simple wordlist checking in combination with a weightening algorythm, every data in range can be decrypted within only a few keystrokes. There is no need to wait for the encryption key to pass from the keyboard to the receier, because it takes only about 20-50 keystrokes to successfully recover the encryption key.

As one can see during one of our presentations and/or our demonstration video, the application basically consists of a sniffer/decoder running in a terminal. As soon as it has estimated the correct encryption key and/or sniffed the valid encryption key, a window pops up and shows all keystrokes from a keyboard in clear text. Every keyboard in range gets its own window. Multiple keyboards can be sniffed at the same time but the quality depends on the radio media and not the application itself.

Patches & Protection Anyone?

To our knowledge there is no patching possibility available for the involved hardware components (Receiver and keyboard). We dont know of any software based encryption solution from Microsoft so far, so you should contact the vendor for help :-).
A bluetooth keyboard might partially enhance the security because the successful sniffing of keystrokes involves eavesdropping of the full pairing sequence. But this could be a wrong assumption, because some vendors might now implement bluetooth proteciton features correctly.

Logitech has additional software that seems to add another layer of encrpytion on top of the communication channel. In addition they promote their “Secure Connect” solution, which should have encryption enabled according to their whitepaper.

How We Did It

Describing the whole process of signal analyzing would extend this document by multiple pages but because we believe the process and analysis of an unkown radio based data signal could be very interesting to others, we put our findings, advisories, procedures and pitfalls into a presentation which we will present at various events in 2008, and will also make it publicly available.

Contact Information

Feel free to contact the authors via email at:

Max Moser	Philipp Schroedel
max.moser@dreamlab.net or mmo@remote-exploit.org	philipp.schroedel@dreamlab.net or psc@remote-exploit.org

About Dreamlab Technologies

Dreamlab Technologies Ltd. is a IT Solution Provider specializing in Information Security and Software Engineering. Since its inception in 1997 Dreamlab provides leading edge IT consulting and highend solutions based on best-in-class Open Standard technologies. For more information please visit http://www.dreamlab.net.

About remote-exploit.org

The team remote-exploit.org is a group of people from various parts of the world, who like to experiment with computers. The website is well known for their uniqe software releases as well as for the world's leading Linux distribution focused on penetration testing, BackTrack. For more information please visit http://www.remote-exploit.org.