IOActive Security Advisory

Title	
Severity	Critical
Discovered by	Mike Davis
Advisory Date	April 30, 2015

Affected Products

1. CyberLock CyberKey based access control solutions.

Overview

CyberLock offers a line of “high security” locks and cylinders as well as related products and services for updating, managing, provisioning, and storing CyberKeys. In various marketing materials, CyberKey is described as “unclonable” and suitable for use in money handling and critical infrastructure systems as a secure and auditable solution.

However, after some reverse engineering it appears that these devices are easily cloned, and new keys can be created from lost cylinders and keys regardless of the permissions granted to the key. Additionally, time-of-day restrictions are enforced by the key, not the cylinder, allowing an attacker access at any time regardless of the configuration.

Issues

1. By intercepting communications between any previously authorized CyberKey and any CyberLock, the site key can be extracted from the lock and used to create cloned keys.
2. All profile-based restrictions (time of day, one time access) are based entirely on the logic of the key itself and can be modified by an attacker who has produced a clone key.
3. While the audit trail stored by the lock provides some information in the event of unauthorized access, the lock itself is not aware of which key IDs are valid. Thus, an attacker can fill the log with nonsensical accesses.
4. Site keys can be recovered from cylinders and are stored in cleartext.
5. The “encryption” (encoding) algorithm used does not sufficiently protect credentials or enforce authenticity.
6. CyberLock cylinders can be relatively easily removed from CyberLock-branded padlock enclosures. With a few sharp strikes to the mechanical retainer, it will shear off, allowing the cylinder to be removed and shackle unlatched.

7. While the issue has already been reported, we feel these issues are exacerbated due to the already known “magnet” bypass (https://www.youtube.com/watch?v=YfldDq48I9U).

Encryption Algorithm

```
CODE:01E9 decrypt_super:                          ; CODE XREF:
CODE:0133^Xp
CODE:01E9                 movlw   (byte_DATA_58)
CODE:01EA                 movwf   BANK0:FSR
CODE:01EB                 movlw   8
CODE:01EC                 movwf   arg_0
CODE:01ED                 swapf   byte_DATA_71, w
CODE:01EE                 addwf   byte_DATA_71, w
CODE:01EF                 subwf   byte_DATA_72, w
CODE:01F0                 addwf   byte_DATA_73, w
CODE:01F1                 addwf   byte_DATA_74, w
CODE:01F2                 subwf   byte_DATA_75, w
CODE:01F3                 subwf   byte_DATA_76, w
CODE:01F4                 movwf   byte_DATA_26
CODE:01F5 loc_CODE_1F5:                           ; CODE XREF:
decrypt_super+14j
CODE:01F5                 movfw   BANK0_INDF
CODE:01F6                 movwf   byte_DATA_25    ; store the
byte
CODE:01F7                 movfw   byte_DATA_26    ; get our key
ready
CODE:01F8                 subwf   BANK0_INDF, f   ; subtract our
key from the byte
CODE:01F9                 movfw   byte_DATA_25    ; get our now
modified byte back into w
CODE:01FA                 movwf   byte_DATA_26    ; store our
newly calculated byte as the key for the next byte
CODE:01FB                 incf    BANK0:FSR, f    ; move on to
the next byte
CODE:01FC                 decfsz  arg_0, f
CODE:01FD                  b       loc_CODE_1F5
```

Technical Details

The CyberLock uses a standard memcmp-style key comparison function, which internally checks the 64-bit site key after “decryption.” As the speed of the response to an invalid key (error code 0x10) is primarily dependent on the number of characters correctly matched, it may be possible to extract the site key through a brute-force attack on a CyberLock in-situ. In (limited) practice, however, it appears that accesses to flash memory adds significant noise to this timing signal and an attack would need to gather enough samples to average these signals out. Additionally, the CyberLock goes into an infinite loop, and the lock itself must be de-powered between attempts, somewhat mitigating this issue.

As the CyberLock is directly powered through the communications port, it appears that an SPA (power analysis) attack may succeed against a CyberLock in-situ, as the lock leaks a significant power side-channel to any potential “key” as the processor slowly clocks the key across an I^2C bus at the Fcpu/4 bps.

However, this approach seems somewhat overboard given the existing issues.

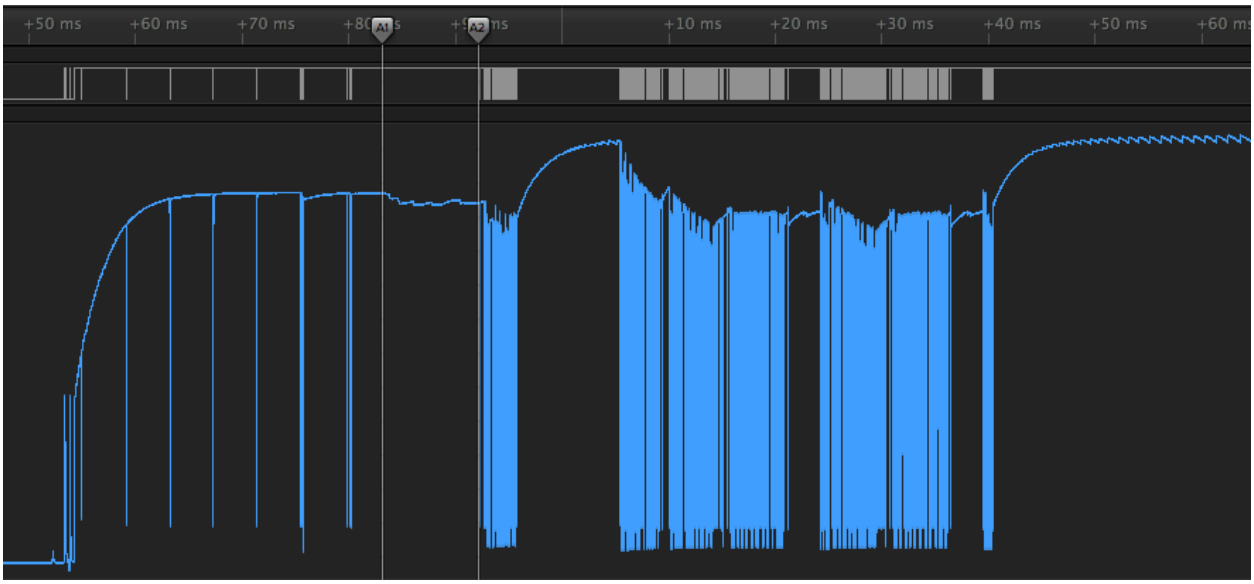

Figure 1: Power draw of initial flash access between marker A1 and A2 during key load (A2 marks the start of the lock ID message 0x02)

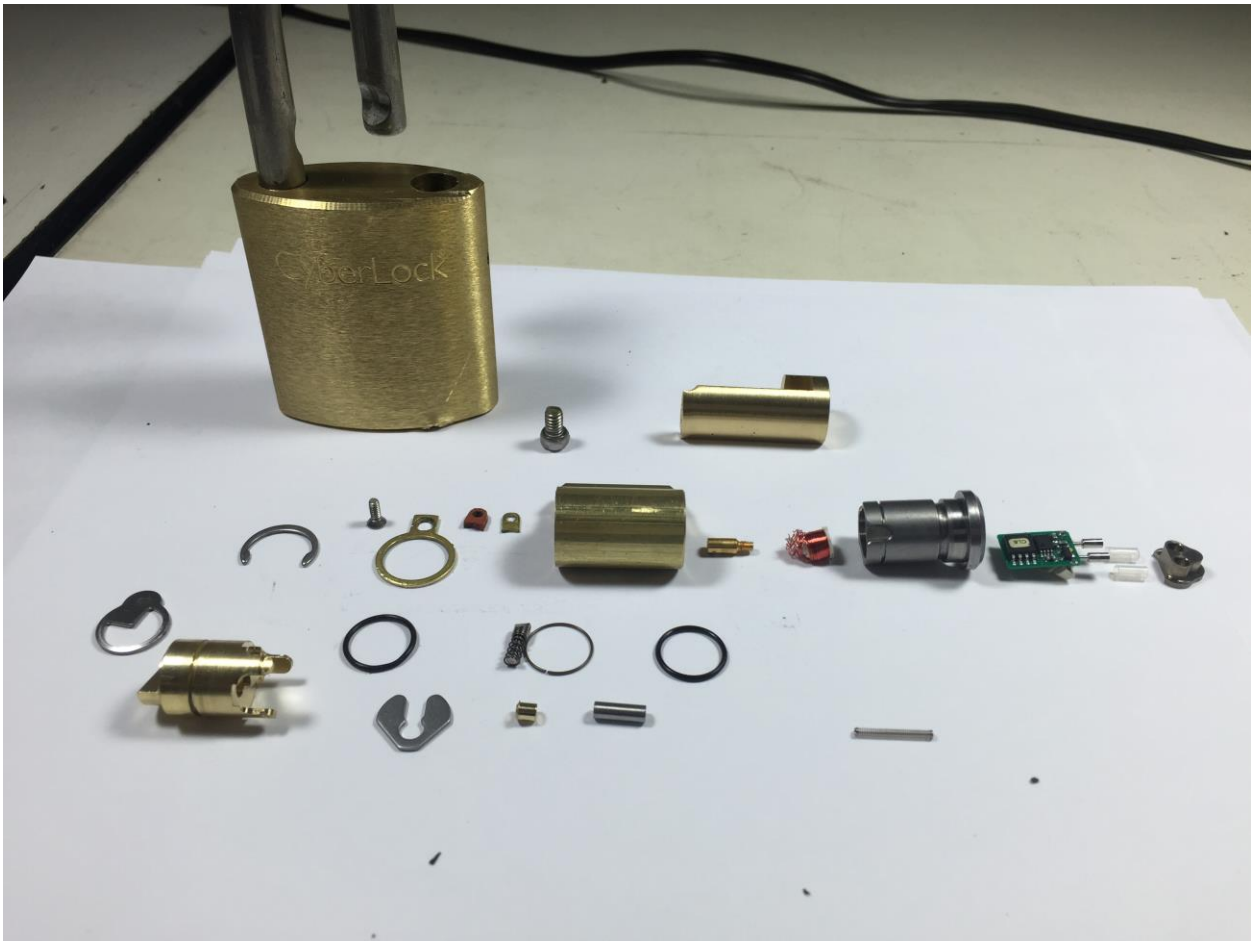

Figure 2: Disassembled CyberLock

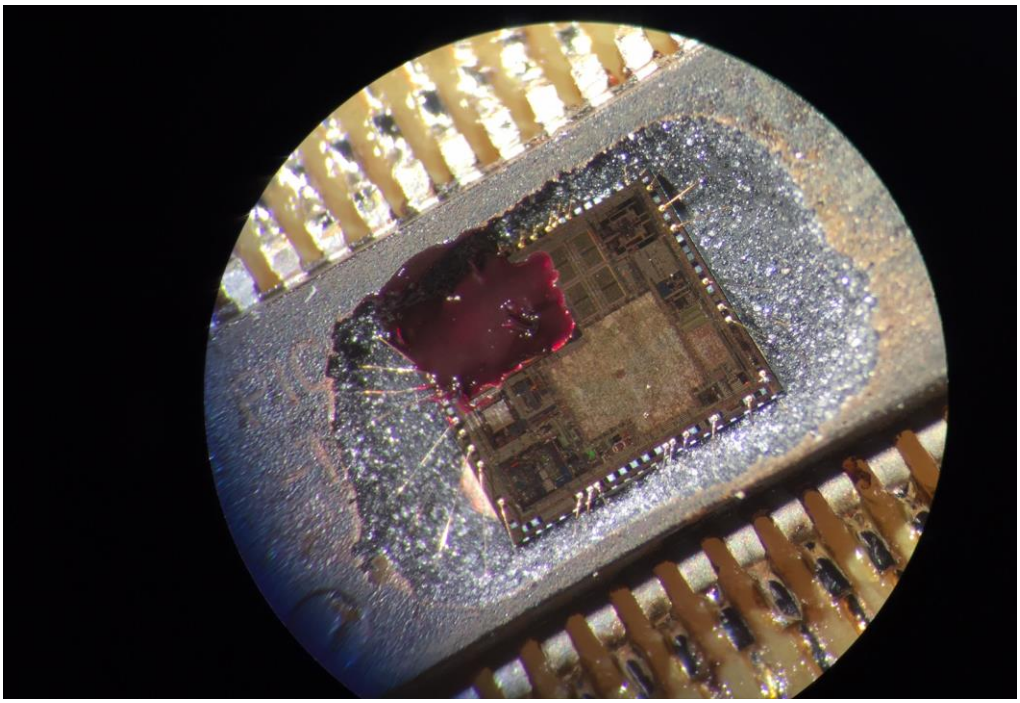

Figure 3: CyberKey firmware extraction methodology

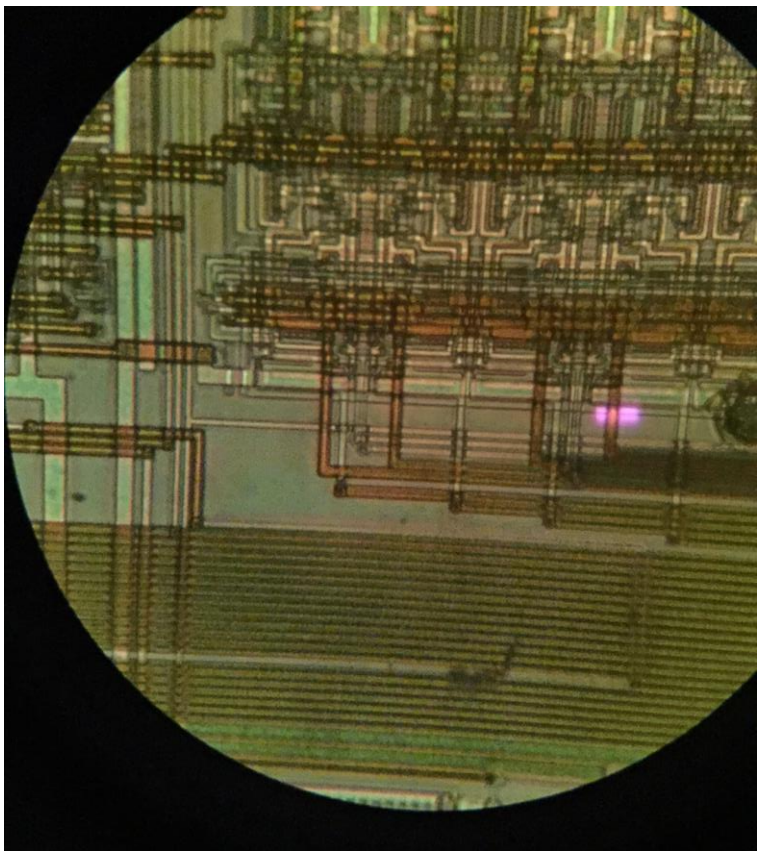

Figure 4: CyberLock fuse clearing methodology for firmware extraction

Responsible Disclosure

- Research period: January 17, 2015 to March 15, 2015
- Vulnerability discovered: March 15, 2015
- Findings finalized: March 30, 2015
- First notification: March 31, 2015 – to Bruce Stephenson, Senior Security Engineer in R&D
- Second notification: April 1, 2015 – to Support@cyberlock.com
- Third notification: April 9, 2015 – to CyberLock sales
- Fourth notification: April 11, 2015 – to Tammy (media relations contact) - *Email delivery confirmation received*.
- Fifth notification: April 17, 2015 – to CyberLock sales and support
- Sixth notification: April 19, 2015 – to CyberLock support

