µBeR: A Microkernel Based Rootkit for Android Smartphones

Joana M. F. da Trindade, Cuong Pham, Nathan Dautenhahn - University of Illinois at Urbana-Champaign

Motivation

Smartphone technology is widely deployed and used daily for a plethora of activities, including banking, email, and social networking.

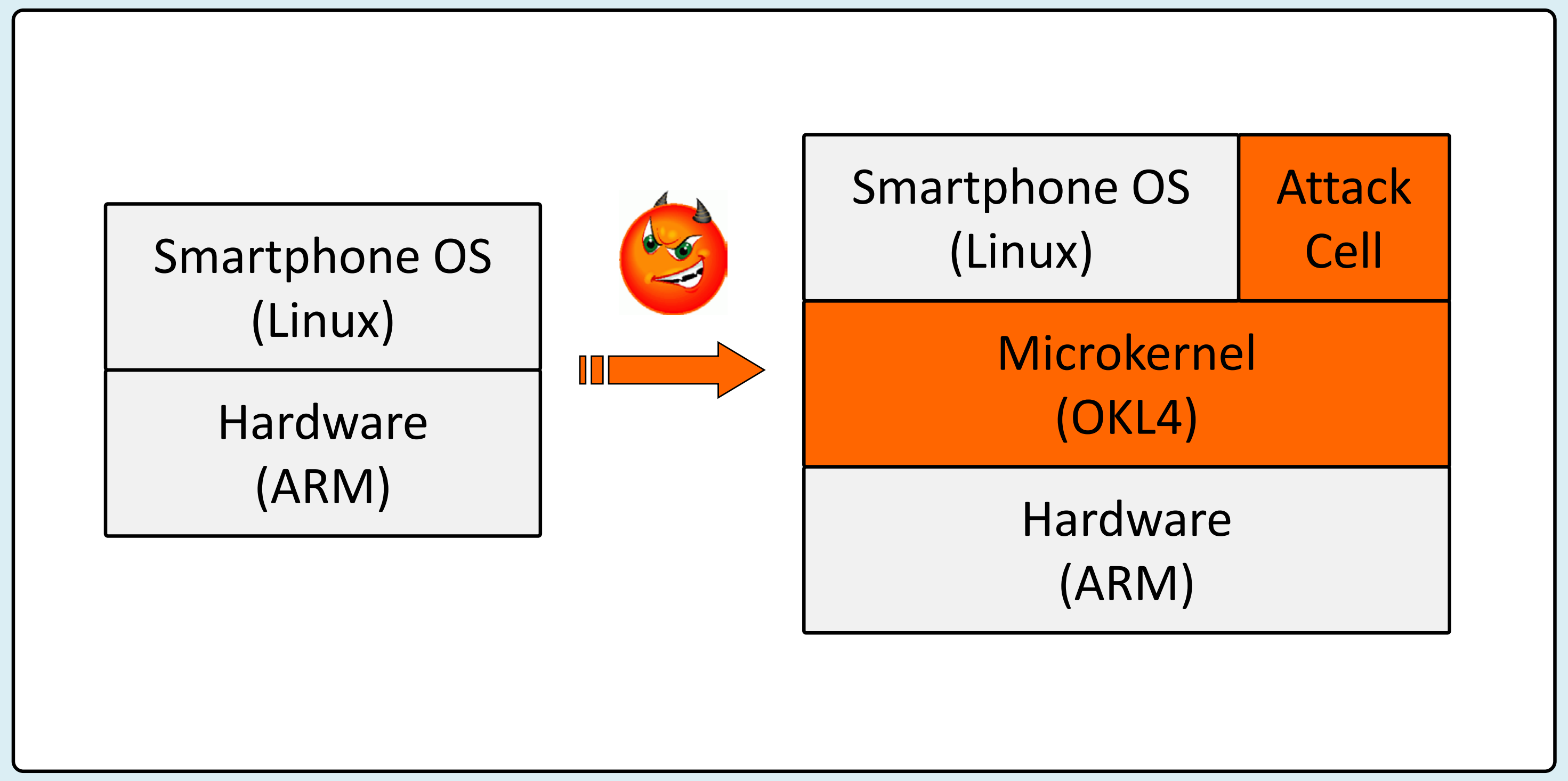
They are also a **desirable target of attack**:
- Not built with security in mind, in part because of resource constraints.
- Store sensitive data and credentials.

Goal

Demonstrate the feasibility and consequences of low-level rootkits on Android (or Linux-based ARM) smartphones.

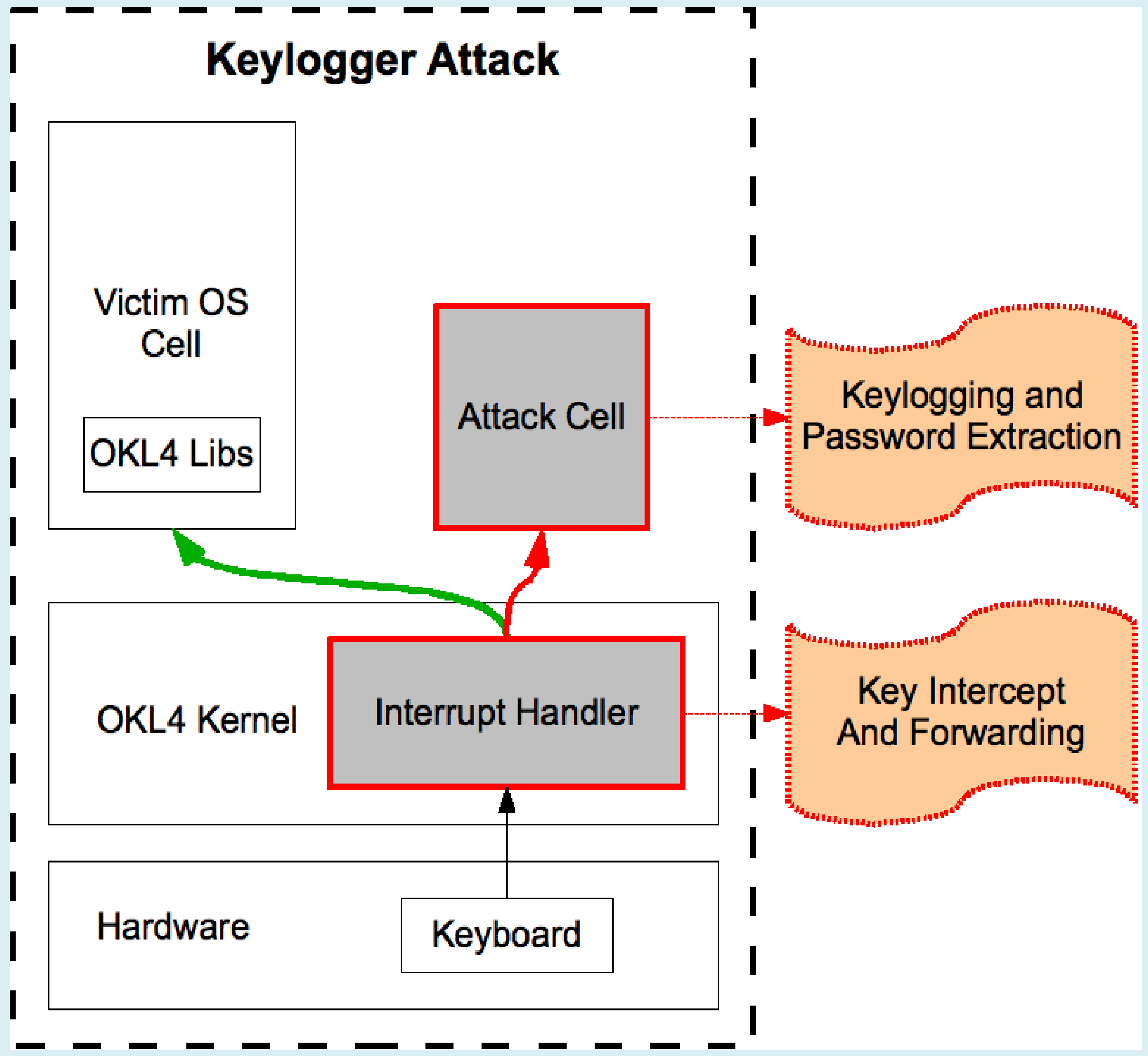
Approach: µBeR, a **Microkernel-Based Rootkit** that leverages **virtualization** technology to hide itself from a target victim OS.

µBeR Design



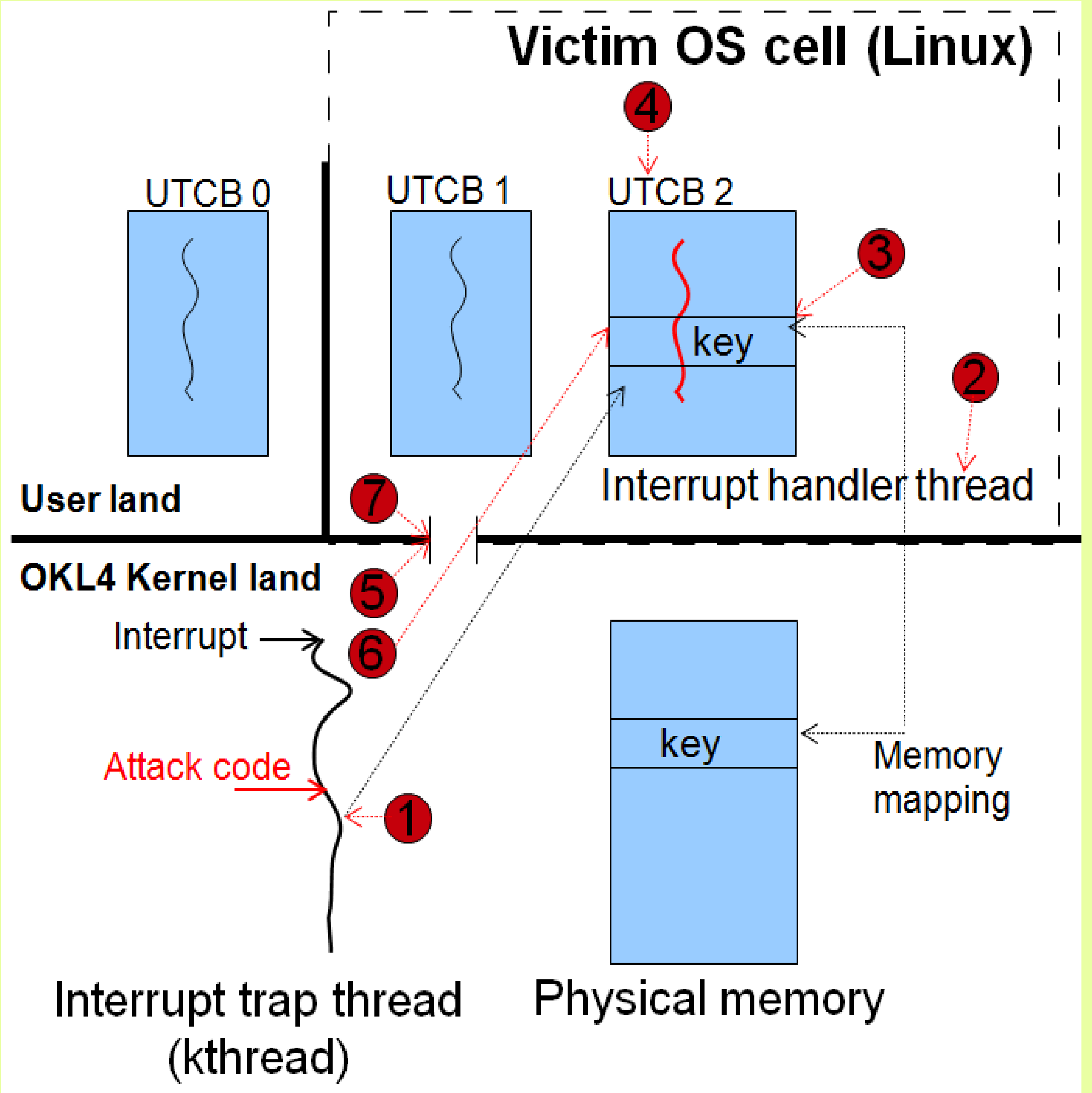
- Rootkit hides itself by inserting a malicious virtualization layer underneath smartphone OS. Auxiliary attack code can also be placed in in a separate virtual machine.
- Similar to Virtual-Machine Based Rootkit (VMBR) approach in SubVirt [1]
- OKL4 Microkernel has smaller footprint than existing Virtual Machine Monitors (VMM) for ARM (e.g., Xen, KVM).
- Goals:
  - Persistent rootkit
  - Undetectable: Keep Smartphone OS state unchanged from OS viewpoint

A Sample Attack: Keylogger



Design Rationale
Apply VM introspection to intercept events on victim OS, and forward data to attack cell.

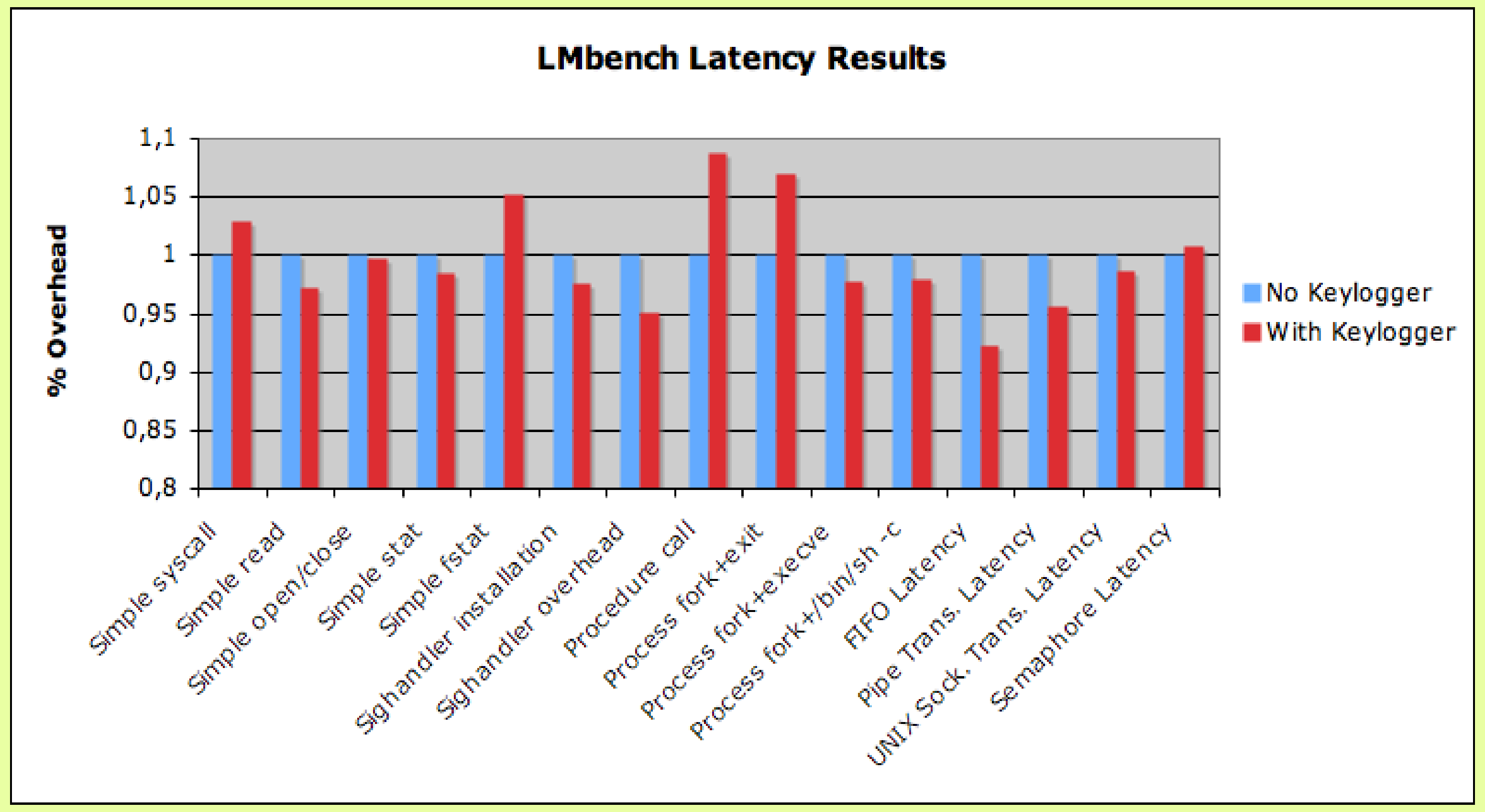
Keylogger Attack Control Flow



Attack Steps

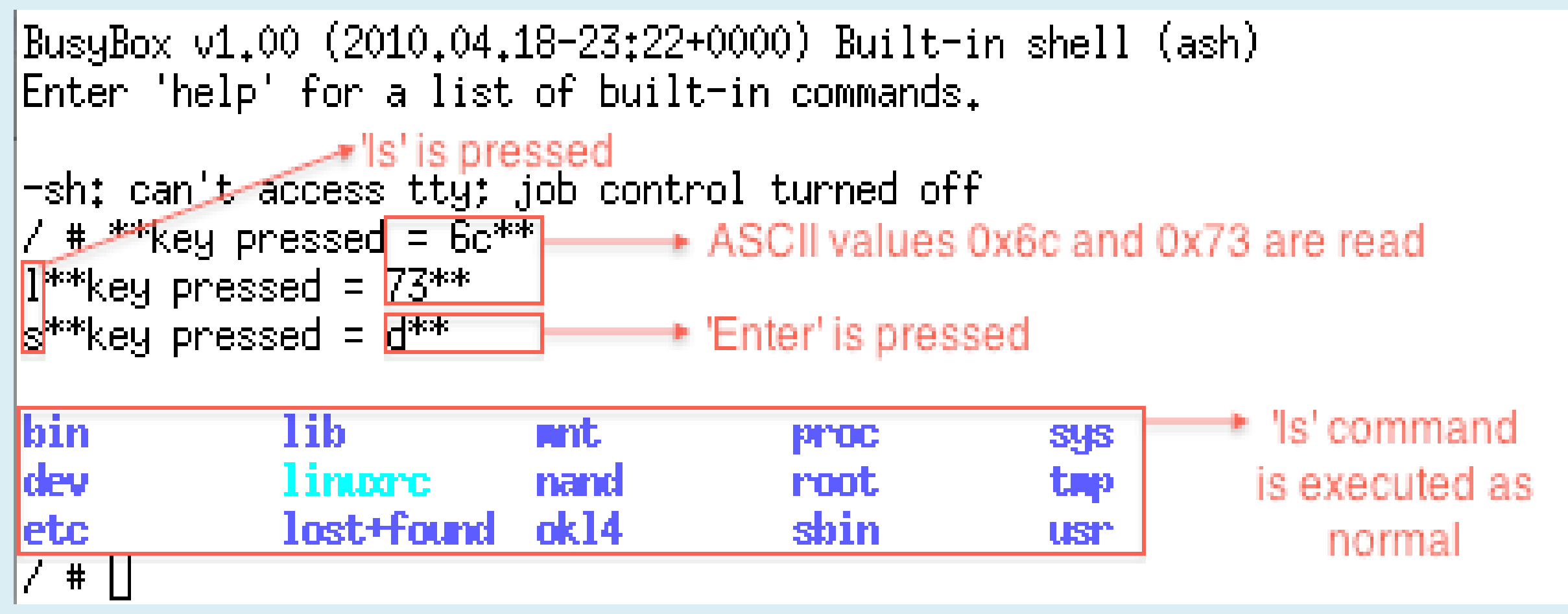
1. Trap key pressed IRQ.
2. Find key interrupt handler thread.
3. Find virtual address of key pressed in user space.
4. Activate user space handler thread.
5. Set user space memory access bit.
6. Read key ASCII data from user space handler thread.
7. Reset user space memory access bit.

Temporal Intrusion



Performance in two configurations: OK Linux + pristine OKL4 3.0 (No Keylogger), and OK Linux + malicious OKL4 3.0 (With Keylogger). Results show less than 10% temporal intrusion in the worst case, which corresponds to 3ms.

µBeR Keylogger in Action



```
BusyBox v1.00 (2010.04.18-23:22+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

-sh: can't access tty; job control turned off
/ # **key pressed = 6c**
l**key pressed = 73**
s**key pressed = d**

bin         lib         mnt     proc    sys
dev         linuxrc     nand    root    tmp
etc         lost+found  okl4    sbin    usr
/ # 
```

'ls' is pressed
ASCII values 0x6c and 0x73 are read
'Enter' is pressed
'ls' command is executed as normal

Initial Prototype uses OKL4 3.0 and a QEMU ARM11 emulator, and prints key data directly to console.

Future Work

- Design and implement a file exfiltration attack.
- Demonstrate feasibility of attack by deploying it into a real Android device.
- Propose mechanisms for detecting and preventing such a rootkit. Existing detection techniques include:
  - Time analysis: OKL4 can spoof clock.
  - VMM detection: not indicative of rootkit.

References: King et al., "SubVirt: Implementing malware with virtual machines," IEEE Symposium on Security and Privacy 2006.