Section A – Personal Data:

a. Name: PanGu Team

b. Photo:

Section B – Training Class Data

a. Title of Training Class:

Advanced iOS Kernel Exploitation

b. Brief Description of Training Class:

This training is designed to teach advanced exploitation techniques for 64-bit iOS kernel. Equipped with seven real world kernel vulnerabilities that most are directly exploitable inside the container sandbox, students will benefit from an in-depth analysis of iOS kernel security features, and learn how to write complete exploits on iOS 9 and iOS 10 for most common kernel vulnerability categories such as info leaks, UAF (Usef-After-Free), race condition, and heap overflow. Some of the vulnerabilities discussed in this training were privately fixed by Apple without public disclosure.

c. Pre-requisite of Training Class:

i. Student :

should be experienced in regular exploit development and understand the root causes of common software bugs. Familiarity with ARM32/64 assembly, and familiarity with C/C++ or Objective-C programming on jailbroken devices are highly recommended

ii. Hardware：

Bring mac laptop with latest Xcode installed
Prepare IDA Pro or Hopper for disassembler

d. Daily Class Outline:

Day 1:

* iOS Security and Development Basic

* iOS Kernel Reverse Engineering 1. iOS Kernel Attack Surface

2. Mach Message Internals 3. How to analyze iOS kernelcache

Day 2:

* Kernel Exploit Technologies 1. iOS kernel memory management 2. Heap Fengshui

3. Return-oriented programming (ROP) * Kernel Exploit Mitigations

1. Evolution of iOS kernel security 2. Hardware based security solutions

Day 3:

* Info Leaks and Heap Overflow Based iOS Kernel Exploitation 1. Vulnerability analysis

2. Exploitation discussion

3. Construct POC

4. Construct controllable heap context

5. Complete exploits step-by-step

Day 4:

* UAF Based iOS Kernel Exploitation 1. Vulnerability analysis

2. Exploitation discussion

3. Construct POC

4. Construct controllable heap context

5. Complete exploits step-by-step

Day 5:

* Race Condition based iOS Kernel Exploitation 1. Vulnerability analysis

2. Exploitation discussion

3. Construct POC

4. Construct controllable heap context

5. Complete exploits step-by-step

* Kernel Patch and KPP Bypass

1. KPP implementation 2. How to bypass KPP