PS4 Hack

2/15/17

* Ps4
  + Released nov 2004
* How to hack
  + Web browser
    - Load your attack/code on site
    - Already run javascript code
* Attck background
  + By cturt
* CVE-2012-3748
  + Vulnerability in WebKit javascript engine
  + Arrays have sort method
  + Array changes while sort is running, the sort routine will write past the end of the array
    - Sort can use custom compare code
    - Custom compare code can change array
  + Why do we care about writing past the end of array
    - Put code you want in the space
* Attacking ps4
  + Point web browser to exploit
* Return oriented programming (rop)
  + Construct a fake stack
  + Stack contain data values and pointer for where to return to
  + Find gadgets from web browser and loaded libraries
    - Bits of code which end with return( or jump if we can control jump target
* JuSt-ROP
  + Write javascript which writes ROP for you
  + Reboot browser and write new program
  + Kinda slow
    - Faster than manual ROP
* Address Space Layout Randomization(ASLR)
  + Used by web browser
* Pwned
  + ROP still restricted by OS of ps4
  + Runs small programs in browser
* What can we do with ROP?
  + Make sys calls
  + OS based on BSD 9, but has custom sys calls
* Gatting past w^x
  + Certain processes must be able to compile and then run code
    - Like web browsers
  + 2 custom sys calls
  + Sys\_jitshm\_create
  + Sys\_jitshm\_alias
    - Together create two virtual pages
* Solution
  + Encrypt the connection
* Further Privcy risk
  + How to know that the tv is legit?
  + A rogue tv device(or device pretending to be a tv) could still record stream
  + Authentication
    - Ensure tv has been authorized by the HDCP people before
* Funky proprietary thing
  + Interesting attempt at cryptography
  + Cant really be called public key or secret key
    - No single shared secret amongst all devices
    - Has a public/private pair
  + Ea device has a 40-bit ID (20 1’s and 20 0’s) which acts as public key
  + Each device has vector of 40 56-bit numbers which acts as private key
  + They exchange IDs and then each one multiplies the other’s ID by their own vector and sums them up
  + Problems with this
    - Means all key-pairs have to be mathematically related to each other
  + Flaws
    - If you have 40 public/private pairs, you can derive a master key
      * Can be used to make new public/private key pairs
      * Thus defeats the authentication
    - Random numbers weren’t really random (based timing between power and start of authentication)
    - Use stream cipher, so you can get the same session Key
* Improvements
  + Dropped whole funky proprietary thing
    - Started using RSA, ESA
    - Very standard
* Why implement HDCP MITM?
  + HDCP restricts the implementation of legitimate content manipulation
    - Picture in picture
* Goal
  + Consumer-side content remixing