CVE-2019-18988 : Setup

* 이 문서는 CVE-2019-18988 exploit을 직접 실행해보고, 이를 통해 어떻게 작동하는지, 찾아낸 exploit이 어떻게 실제로 활용될 수 있는지를 시연하기 위한 가이드라인으로 작성되었습니다. 작성자도 전 과정을 실행해 보고 확인한 것이 아니기 때문에, 중간에 문제가 발생할 수 있습니다. 경험상, 이미 나와 있는 exploit을 실제로 확인하고 실행해 보는 데만도 상상 이상으로 시간이 많이 들어갑니다.

Storyline

현재 계획은, 다음과 같은 상황을 가정합니다.

- Attacker는 Victim의 머신에 Teamviewer 7이 설치되어 있다는 사실을 알고 있습니다.
- 양쪽의 PC에는 python이 설치되어 있습니다.
- 여러 방법을 사용해서 (이메일 등...) Attacker가 Victim의 머신에 exe 파일을 전송, 이를 실행하도록 유도합니다. 이 프로그램은 Victim의 Registry에서 Teamviewer password를 추출, 이를 Attacker에게 전송합니다.
- Attacker 측에서 미리 작성한 다른 스크립트가 이를 정리하여 Attacker가 Victim의 팀뷰어 정보를 볼 수 있습니다. Teamviewer에는 이 외에도 알려진 여러 취약점이 있으며, 이 Teamviewer password와 다른 취약점이 결합하여 원격 조종 권한의 탈취, 도청 및 화면 강제 공유 등 여러 추가적인 attack이 가능합니다.

가상 머신의 Windows가 Victim 역할을, 호스트 Windows가 Attacker 역할을 수행하게 됩니다.

Requirements : VM / Teamviewer

* 이외의 환경에서 확인해보지 않았습니다. 성공하면 제보(?) 부탁드립니다. (이하 모든 항목)

- Virtualbox를 설치해 주세요. 이번 뿐 아니라 앞으로도 가상 머신은 지속적으로 사용됩니다.[1]
- Windows 10 VM을 준비해 주세요. [다운로드 링크] 에서 Virtualbox용 이미지를 선택해 주세요.
- VM을 설치합니다. Windows 쪽의 RAM을 최소 4GB 이상 할당하기를 권장합니다.
  설치 방법은 여기저기 많이 나와 있습니다.
- Teamviewer 7을 설치합니다. [다운로드 링크] [2]
- Teamviewer를 실제 사용하는 것처럼 필요한 설정들을 해야 합니다. password 세팅하는 칸에 전부 설정해 주세요.

Requirements : Environment

- 역할을 할 머신 양쪽 모두 python을 설치해야 합니다.
- Victim 역의 python에는 다음의 package가 설치되어야 합니다. 이를 pip을 통해 설치해 주세요.
  `winreg json codecs`
- Attacker 역의 python에는 다음의 package가 설치되어야 합니다.
  `binascii hexdump pycrypto`

[1] 개인정보를 탈취하거나, 뭔가를 무너트릴 목적으로 개발되었으며, 누가 작성했는지 모르고, 그 작동을 완벽하게 이해하지 못하고 있는 코드를 지금 쓰고 있는 machine에 설치해서 돌려볼 용기가 있으신가요?

[2] 버전 7일 필요는 없습니다. 7부터 13까지는 작동하는 것으로 알고 있습니다.

What this is

CVE-2019-18988의 핵심은 두 가지입니다.

- Teamviewer는 모든 Password를 단일 AES Key와 IV를 이용하여 암호화합니다. 여기서 모든 Password라 함은, Option password / Account password / Unattended access password 등 팀뷰어의 기능 각각에 걸려 있는 여러 password들을 의미할 뿐 아니라, **모든 유저의** 패스워드를 의미합니다. 즉, 전 세계의 모든 팀뷰어 유저 각각이 쓰고있는 모든 팀뷰어 프로그램을 통틀어 AES Key / IV Pair는 단 하나만 존재합니다.
- 암호화된 값은 레지스트리에 저장됩니다. 레지스트리 자체는 python이나 ruby 같은 고수준 언어로도 어렵지 않게 접근할 수 있으며, 전체의 json dump를 뜨는 것도 어렵지 않습니다.

2번 항목은 사실 그렇게까지 심각한 일은 아닙니다. 암호화된 값을 레지스트리에 저장한다는 것은 어떻게 보면 당연하고, 굳이 암호화된 값을 열심히 방어해야 할 이유는 없습니다.
1번 항목은 있을 수 없는 일입니다.
이 취약점 최초 발견자의 원본 글에서 이 AES Key를 찾아낸 방법은, Teamviewer 전체의 실행 파일을 Reversing하는 것입니다. Reversing은 상당히 어려운 기술이며, 수천 수만 줄의 Assembly code 또는 binary를 읽어야 하고, 적절한 툴이 필요한 아무나 할 수 없는 방법입니다. 그러나 누군가 작정하고 특정 Instance를 상대로 reversing한다면 막을 방법이 마땅치 않은 것도 사실입니다. [3]
만약, 모든 유저에 대해 다른 key/iv를 이용해서 암호화했거나, 적절한 대응을 수행했다면 2번 자체는 문제가 되지 않습니다. 특정 유저를 타겟팅해서 그 유저의 AES key를 알아내는 작업은 어려울 뿐 아니라, 이미 그 과정에서 그 머신에 대한 매우 높은 수준의 control이 요구되므로, (실행 중인 프로그램의 memory dump 등) 그정도 control이 있다면 팀뷰어 비밀번호를 알아내야 할 이유가 별로 없습니다.

Attack module : How it works

매우 간단한 이 코드의 기본적인 원리는 원본 글 후반부의 Ruby code (metasploit module) 과 같습니다. 이 글에서는 metasploit이라는 매우 널리 쓰이는 해킹 툴을 이용, 그 툴 위에 모듈로 붙이기 위해 작성되어 있지만 우리는 metasploit을 사용할 계획이 아니므로 별로 상관은 없습니다. 다만 이런 경우 Victim machine에 실행되는 결과를 Attacker가 받아와야 하고, 이부분을 어떻게 처리해야 하는지에 대한 issue가 남습니다.

Attack module : What we want

- 이 코드를 exe 파일로 만들어서, Victim이 one-click으로 실행하게 하고, 가능하다면 victim이 무엇이 실행되었는지 알기 어렵게 할 수 있을까요?
- 현재는 구현해두지 않았는데, Victim의 머신이 직접 네트워크 등을 통해 Attacker에게 스스로 password를 전송해 주도록 할 수 있을까요?

[3] 이 기술은 게임 실행 파일을 뜯어보는 데 정말 많이 사용됩니다. 혹시 포켓몬 게임에 대해 아신다면, 1세대 게임의 수많은 버그에 대해 들어보셨는지 모르겠습니다. 특히 콘솔 게임의 버그성 플레이 / 숨겨진 요소들이 이렇게 찾아졌습니다.