IOT僵尸Mirai源码浅析

2016-11-17 泰格实验室 [泰格实验室](http://mp.weixin.qq.com/s?__biz=MzAwMTk5MzEwNw==&mid=2247483716&idx=1&sn=e3053c732e724ba1aa3064234ca0de0d&chksm=9ad071b8ada7f8ae7e5ad1025909308145c38a96b60b77914b91282284af0b35909458ec69bf&mpshare=1&scene=1&srcid=1117DK0KzENwVkQ0YkhMuAMR##)

Mirai僵尸网络在近期针对美国网络服务商Dynamic Network Services Inc的大规模分布式DDoS攻击中扮演了重要角色。mirai可以高效扫描物联网系统设备，感染采用出厂密码或弱口令的物联网设备，物联网设备被感染后成为僵尸网络bot，最终形成一个庞大的僵尸网络，随后在黑客命令下发动DDoS攻击。由于mirai作者将木马程序源码在github上进行了公布，我们对此源码进行了分析验证。

**一、Mirai简介**

        木马程序mirai是基于Linux系统的恶意程序，主要针对网络摄像头、路由器等物联网设备进行感染，支持x86、arm、m68k、mips、ppc等多种架构。该木马程序主要包括CNC管理端、bot端、loader和tools四部分。CNC是木马控制程序，僵尸网络管理员及用户可据此管理僵尸网络；bot端为木马程序主体程序，主要有DDoS攻击、设备扫描、扫描信息上报、防设备重启等功能；loader的主要功能是木马感染，通过弱口令进入物联网设备，然后将木马主体植入并执行；tools主要包含enc、scanlisten、wget等多个小程序，作为附加工具，方便用户配置管理僵尸网络。

**二、Mirai网络结构**

        在mirai僵尸网络中主要有管理员、用户、CNC服务器、loader服务器、下载服务器和bots（物联网设备）。僵尸网络结构图如图1所示。

 图1. mirai僵尸网络结构图

1. 管理员：僵尸网络的创建者和管理者，通过telnet登录CNC服务器，可以创建用户，设置用户名、密码、攻击时长、可使用bot数、账户可登录天数等内容，同时也是整个僵尸网络的直接使用人，可以控制全部bot进行DDoS攻击。笔者推测，管理者根据用户支付的钱数，为创建账户并分配bot资源。
2. 用户：僵尸网络的使用者，通过telnet登录CNC服务器，可以查看自己账户下可用来发动攻击的bot数，管理本账户可用bot进行DDoS攻击。通过向僵尸网络管理员支付金钱购买bot进行DDoS攻击。
3. CNC服务器：僵尸网络的管理服务器，采用mysql数据库，开放23和101两个端口，管理员、用户和bot均通过telnet协议进行通信，101端口为API接口监听端口。
4. loader服务器：扫描结果接收服务器及木马植入服务器，监听48101端口，此服务器通过scanlisten程序监听48101端口，接收bot扫描的弱口令设备信息（IP：port：username：passwd），并传递给loader程序。loader程序根据扫描信息登录物联网设备，并进行架构探测，随后上传木马下载程序，并控制设备下载木马程序并启动木马程序。待将物联网设备感染后，bot继续进行扫描、回传、植入，不断扩大僵尸网络规模。
5. 下载服务器：木马程序下载服务器，通过提供HTTP或FTP服务器供木马下载者下载木马。
6. Bots：网络摄像头、DVR、路由器等物联网设备，僵尸网络的主要受害者，被控制的bot主要进行反检测、防设备重启、DDoS攻击、设备扫描和扫描结果回传等操作。

**三、代码结构**

        Mirai代码主要包含dlr、loader、mirai和script四个文件夹，其目录结构图2所示。

图2. 代码目录结构

**1、dlr文件夹**

        此文件夹下只有main.c一个文件，通过交叉编译环境可生成多个版本dlr文件，主要功能是通过HTTP协议下载木马主体程序，在物联网设备感染过程中扮演下载者角色。

**2、loader文件夹**

        此文件夹下是木马植入程序代码，由于运行在指定服务器上不需要交叉编译环境，编译生成loader可执行文件，主要工作是通过telnet的方式登录bot扫描出的具有弱口令的物联网设备，然后上传木马下载者程序，执行下载指令，最终将木马程序植入到此物联网设备上。

**3、Mirai文件夹**

        mirai文件夹分为三部分内容，CNC主控端部分、bot部分和tools工具包部分。

**CNC：**

        CNC为僵尸网络管理程序，由go语言编写，可在go环境下直接编译。生成CNC可执行文件，运行此程序监听23端口和101端口。Bot通过telnet回连23端口并注册及获取攻击命令。管理员或用户通过两种方式进行管理：（a）使用telnet通过用户名密码登录的方式连接23端口，并进行管理；（b）使用apikey通过101端口控制僵尸程序，此端口推测是为了方便用户使用程序对僵尸程序进行管理。具体来说，cnc文件夹代码功能如下：

（1）admin.go ：用户管理模块，用户通过Telnet23端口连接至此，并对僵尸网络进行管理，主要包括创建用户、查询僵尸、下发攻击任务等功能；

（2）api.go ：API接口模块，用户通过apikey连接101端口至此，其功能与admin模块类似；

（3）attack.go ：攻击模块， admin和api模块调用该模块来生成攻击命令，该模块再调用database模块进行任务下发等操作；

（4）bot.go ：bot模块，主要完成新bot的注册；

（5）clientlist.go ：bot管理模块，主要功能完成bot列表的管理，增、删bot等功能。

（6）datebase.go ：数据库管理模块；主要是对数据库进行管理，用户管理、登录管理、apikey检查、任务启动等，主要管理users、history、whitelist三个数据库表；

（7）constants.go ：配置文件；

（8）main.go ：程序入口；

**Bot：**

        bot文件夹下为bot程序的主体代码，通过c语言编写，利用作者提供的交叉编译环境可进行编译，其debug版本有调试信息、且只能进行一次DDoS攻击，release版本为实际运行版本，具有自删除、进程隐藏等功能。Bot程序主要实现以下功能：

        （1）安全防护

        文件运行后自删除；阻止linux watchdog重启设备；往虚假地址进行连接，欺骗分析者，增大分析难度；回连地址加密存储；监听127.0.0.1：48101端口，如果该端口被占用，则终结该进程，保证同一时刻只有一个实例运行。

        （2）进程隐藏

        采用进程名字随机化的方式来隐藏僵尸程序自身进程。

        （3）防止机器重启

        关闭ssh（22）、telnet（23）、http（80）等服务，防止其他用户登录此机器。

        （4）设备扫描

        使用弱口令对随机地址进行扫描探测，发现有弱口令机器则上报到loader服务器。

        （5）攻击任务

        Bot端支持9种DDoS攻击方式，具体攻击方式如下：

        udp: UDP flood

        vse: Valve source engine specific flood

        syn: SYN flood

        ack: ACK flood

        greip: GRE IP flood

        udpplain: UDP flood with less options. optimized for higher PPS

        dns: DNS resolver flood using the targets domain, input IP is ignored

        stomp: TCP stomp flood

        greeth: GRE Ethernet flood

        http: HTTP flood

**Tools：**

        本文件夹下主要是一些攻击用的工具，如回连地址变换、更新工具等。

        （1）enc：回连域名加解密程序；

        （2）scanlisten：扫描结果接收器；

        （3）wget：程序下载器；

        其他几个小程序暂时不知用处。

**Script：**

        此文件夹下包含一个cross-compile.sh配置脚本和db.sql数据库文件。笔者在测试时利用cross-compile.sh配置环境失败，通过网络下载其所需文件并采用其中部分命令配置成功，最终编译出多个版本木马程序；db.sql可直接在数据库中运行，经测可用。

**四、控制命令**

        1、adduser 添加用户命令，可设置用户名、密码、攻击时长、可使用bot数、账户可登录天数等内容。

        2、botcount 查看当前在线bot数

        3、攻击命令，可下发9种攻击命令

        格式为 protocol  target  ip  duration  srcip  srcport  dstip  dport 。

        示例： udp 192.168.1.104 300 dport=10021

**五、数据库结构**

        Mirai程序采用mysql数据库，数据库名为mirai，该数据库主要包括三个数据库表：history、users和whitelist。

        1、表histroy记录了每次攻击的详细信息。包括攻击ID、用户ID、发送攻击的时间、攻击持续时间、以及具体攻击命令、发动的僵尸数量等内容。

        2、表users主要记录用户的一些信息。包括用户名、密码、能够发动攻击的持续时间、付款时间、能够发动的僵尸最大数量、用户类型等信息。

        3、表whitelist记录了一个白名单，在管理端生成攻击任务时会检测攻击目标中是否包含白名单中IP，如果有则会将相应IP过滤掉。

**六、编译环境说明**

        mirai文件夹中的编译脚本为build.sh文件，需配置交叉编译环境，编译范围包括bot、cnc、tool三个文件夹中的c代码和go代码。编译的版本包括debug版和release版，其中debug版有调试信息且编译出可执行文件对应的架构较少，release版为无调试信息的实际感染版本。

        dlr文件夹中内容也需要交叉编译环境，可编译出多个版本可执行文件；

        loader文件夹是文件使用标准gcc编译即可，主要和搭建的服务器环境有关，不需要交叉编译环境。

        由于此僵尸程序需在多种架构中运行，故编译环境为交叉编译环境，环境配置参考1rRCc3aD.sh文件和cross-compile.sh文件，经测试可用。

**关于作者**

泰格实验室是一群来自高校、安全公司和互联网公司的网络安全爱好者组成的安全研究团体。欢迎小伙伴们随时与我们联系，团队官网：www.hashkill.coom，联系QQ:2187178702，QQ群:310685981，E-mail:root@hashkill.com。