fai-project 系统安装定制简介

服务器通用操作系统基础设施常规配置、性能优化、安全加固

本方案系统支持优先级：debian>ubuntu>centos,这意味着低优先级会有更多的错误与未实现功能，实际生产环境服务器会根据情况调整配置，本文档描述与服务器不一致的地方以服务器配置文件为准

前期准备：

1. NTP 服务器 2 台，提供时间服务
2. DNS 服务器 2 台，外网 dns 缓存与内部名字解析
3. ZABBIX 监控 1 台，基础监控
4. ANSIBLE 主控 1 台，自动化运维，提供 ssh 公钥给被控服务器
5. 备份机存储 1 台，保存所有服务器镜像备份，提供 ssh 公钥给被控服务器
6. Rsync 客户端 IP 列表，web 后端更新代码
7. Repo 代理缓存 1 台，提供 repo（同时支持 debian、ubuntu 和 centos）内网缓存
8. 安装过程保持网络畅通，安装时软件包更新需要访问网络
9. Nfs 服务器提供静态 web 文件存储
10. Syslog 远程日志服务器 1 台，集中保存所有服务器日志

常规系统增强与配置变更

1. 无论服务器有多少个网卡，第一个网卡必须为内网网段（管理网络、内部数据网络、存储网络），并且支持 PXE 启动，第一个网卡名字 eth0
2. IP 地址与 MAC 绑定并且 DHCP 分配的地址与静态设置一致,安装完成自动生成静态 IP 配置文件
3. 初始安装是安全的不必担心来自网络的入侵。
4. 所有服务只监听内网地址的端口,禁止监听 0.0.0.0, 仅有对外提供的服务可以监听外网 IP 内部服务： sshd ntpd dns-cacher zabbix mysql-server redis memcache
5. 默认用户 hmuser 使用 32 位字串复杂密码，可以 sudo 至 root
6. Root 用户内置 ansible 控制端公钥可以直接 root 登录, Sshd 服务端 禁止 root 密码登录使用公钥或者普通用户登录后 su 或者 sudo 禁止 dns，Ssh 客户端开启持久连接，加密算法使用椭圆曲线 ecdsa
7. dns 解析，时间同步,日志记录使用内网服务器
8. grub 关闭 splash 网卡使用 ethX 名称， 详细输出，控制台分辨率 1024x768 io 调度器 deadline，关闭光标闪烁，
9. apt-get 修改源（repo）自动使用局域网代理缓存 https 不用代理
10. 包含 sysctl 内核参数调优，sysfs 调优，ext4 文件系统调优（性能调优以真实业务客观化观察计时测量数据为准，而非跑分与基准测试）。
11. 彩色控制台输出 man ls grep
12. 默认语言 en_US.UTF-8
13. iptables-persistent 保存防火墙规则
14. needrestart 更新安装软件检查是否重启
15. smartmontools 监视硬盘 smart 信息
16. nscd 本地名字服务缓存（user，group,hosts,service）
17. apt-show-versions 检查软件包版本
18. apt-fast 多线程快速下载软件包

19. cron-apt 自动检查下载更新，默认不安装任何更新
20. **unattended-upgrades** 自动检查下载安装安全(secure)更新
21. ipwatchd IP 冲突检测
22. 用户默认交互 shell 改为 zsh，bash 仅运行脚本
23. kexec-tools 内核热重启（不经过 bios 自检）
24. 性能观察 sysstat, iftop（查看网卡）, itop（查看中断）, dnstop（查看 dns 解析）, powertop（查看电源）, apachetop（查看 apache）,atop（全面查看进程）, hatop(查看 haproxy),jnettop（查看占用流量最多的主机/端口）, kerneltop（查看内核函数使用，需要内核开启 profiling）, mytop（查看 mysql）, nethogs（进程的网络使用）,htop（查看进程）, pgtop（查看 postgresql）, sntop（检测主机存活）, virt-top（查看虚拟机）,Blktrace（追踪块设备 io）, dnstracer（追踪 dns 解析）, fatrace(追踪打开文件), fxt-tools（追踪多线程）, ioapps（重放 IO 追踪与分析 gui）, kmtrace（追踪内存泄漏 gui）, latrace（追踪动态链接库）, leaktracer（追踪内存泄漏）, lft（四层路由追踪）, mutextrace（追踪互斥体）,mtr（路由追踪）, netsniff-ng（网路包嗅探）, pstack（追踪进程栈）, tcptraceroute（追踪路由）, kernelshark（GUI）, trace-cmd（追踪命令执行）, tcptrack（追踪 tcp 连接状态）, bmon 带宽监控工具, Collectl：一体化性能监控工具, Nmon：监视 Linux 性能,iptraf 网络流量查看,glances 监视系统性能，dstat 系统性能

25. 性能动态追踪 bcc-tools, sysdig（无法在 grsecurity 内核运行）, Lttng-tools,lttv,lttngtop,（Linux 追踪工具）
26. lm-sensors 监视传感器数据
27. cpufrequtils CPU 主频控制
28. irqbalance SMP 处理中断

安全保护系统的设计原则
(1). 最小特权：为使无意或恶意的攻击造成的损失最低，每个用户和程序必须按需使用最小特权；
(2). 机制的经济性：保护系统的设计应小型化、简单、明确，保护系统应该是经过完备测试或严格验证的；
(3). 开放系统设计：保护机制应当公开，理想的情况是将安全机制加入系统后，即便是系统的开发者也不能侵入这个系统；
(4). 完整的存取控制机制：对每个存取访问系统必须进行检查；
(5). 基于“允许”的设计原则：说白了就是“白名单”策略，基于否定的访问控制策略；
(6). 权限分离：实体的存取应该受到多个安全条件的约束；
(7). 避免信息流的潜在通道；
(8). 方便使用友好的用户接口；

系统安全加固
保持更新至最新补丁只能避免已知漏洞的威胁，无法对抗 0day 漏洞，基于特征库的扫描性能开销巨大，同样无法对抗未知攻击，此类工具（方法）仅限于应付已知威胁。
对抗 0day 漏洞的前提基础是有能力解决已知漏洞威胁，公开但是未修复漏洞视作 0day 处理，即使具备对抗 0day 漏洞能力，系统依旧需要做常规安全更新与软件升级（30 天内修复已知漏洞）。

安全防御加固，按照最小权限，主动免疫，多重防御，最低成本，最高收益法则实行，目标具备前瞻性，防患于未然，先发制人，能够有效对抗未知木马、未知 rootkit、未知后门、0day 漏洞、程序 bug、以及人为操作失误的威胁（需要做概念验证，证明其有效，无法验证、验证失败或者拒绝验证认为无效措施），无需改变现有应用，最大限度保证程序按照预期的方式运行。

~~其他措施：监控、审查、取证、保密、隐匿、伪装、反击，震慑不良企图~~

暴露在外网的系统要具备相当的威慑力，迫使攻击者不敢贸然行事，针对来自中国的攻击者，系统威慑力排名如下：BSD>Linux>Windows，这样安全防护人员可以集中精力专注于高级别高技术水准的严重/致命威胁，而不是疲于奔命，应付骚扰与无实质伤害的威胁，避免针对人员的 DOS 攻击。

系统应当具备健壮性，有相当的抗打击能力，部分节点，组件失效或者被摧毁仍然可以提供所需功能。

安全设计应当尽可能小的影响性能，额外的开销尽可能的少，小到无法观察测量，最短的响应时间，根据出现频度不同，监视过滤拦截异常应该在 1 毫秒之内完成，甚至更低的时间。

整个生命周期内尽可能低的复杂度，尽量少的人力开销，便于实施、维护、升级。

堡垒最容易从内部攻破，相对于外部威胁，内部威胁更需严加防范，把服务器上的所有用户都看成是怀有恶意的使用者，所以系统必须限制使用者的权限，并且监视记录使用者行为，这是一个无关道德的问题，而是一个技术能力的问题。

强制访问控制（RBAC 角色访问控制）

~~限制系统里面的用户(包括 root)，必须按照管理规则预期的方式工作，规则之外，完全禁止。~~

~~系统管理特权实行三权分立~~

~~系统管理员（root/adminstrator）系统管理员负责系 统的安装、 管理和 日常维护。 如安装软件 、 增添用户账号、数据备份等 ，类似于公司的总经理~~

~~安全管理员（secadmin）安全管理员负责安全属性的设定与管理。类似于公司的监事会；~~

~~审计管理员（auditor）审计管理员负责配置系统的审计行为和管理系统的审计信息。 类似于公司的董事会 。~~

~~三个角色互相制约。攻击者破获某个或某两个管理角色 的口令时不会得到对系统的完全控制~~

伪装不能实质提升安全水准，但是可以有效阻止扫描器、ROBOT（机器人）、AI（人工智能）自动化攻击，迷惑与麻痹攻击者，加大攻击难度系数，使其犯更多错误，付出更多时间成本与代价。

~~主动防御：主动侦查可能攻击己方的对手，进行防范和报复。~~

~~自毁装置与触发条件设计~~

~~数据机密性保护~~

系统完整性保护，不被篡改，以及被篡改及时发现。

审计与取证，秘密审计不被用户察觉，无法逃脱，无法被清理/毁灭证据

~~限制 Intel ME（*Intel Management Engine*）的同时加强信任链条。~~

注意：无法对抗底层攻击，固件层，hypervisor 层，硬件设备层。

1. 提供 2 类内核，默认普通内核用于安全性要求不高的场景，grsecurity/pax 加固内核用于需要高级别安全特性的场景，内核开启 apparmor，禁用 selinux（斯诺登“棱镜门”之后禁

止一切 NSA 相关代码运行）

2. ~~复杂密码，sha512 散列,密码强度，至少是大小写字母+数字+符号一共 16 位以上（libpam-cracklib）~~
3. 限制 ssh 用户只能内网 IP 登录
4. ~~拒绝 root 登陆远程和控制台以及 su~~
5. 限制 secuetty root 登录
6. ~~禁止加载内核模块，kernel.modules_disabled=1~~
7. 默认用户掩码 077
8. 有且仅有 adm 组用户切换到 root 权限
9. ~~取消 root 用户无限大权限，限制其实际权限等同于或者弱于普通用户，对抗 root 提权漏洞（grsecurity+RBAC）~~
10. ~~系统全局启用基于角色访问控制 RBAC，并且实施最小权限（grsecurity+RBAC）。~~
11. 隐藏 pid，普通用户禁止查看其他用户进程
12. ~~隐藏或伪装操作系统指纹，避免被 nmap 扫描（grsecurity 未完全实现，hardenedbsd 未完全实现），修改 tcp 协议栈，避免被探测开机时间。~~
13. ~~应用沙箱 firejail，fakeroot-ng~~
14. 禁止 usb 存储设备，firewire 存储设备，禁止 pc 喇叭，禁止蓝牙设备
15. 启用进程记账 acct，~~启用网络记账 net-acct(网卡开启混杂模式，记录所有 IP 连接，包括对外发起连接和各种扫描器连接失败)~~
16. 禁止每日备份密码文件 passwd，group,shadow,gshadow
17. 禁止安装 locate 和 updatedb 避免泄密
18. ~~Arpwatch 监视 arp 攻击~~
19. Debsums 检查已安装软件包完整性
20. Debsecan 检查 CVE 漏洞报告
21. 只读挂载共享内存/dev/shm（会影响部分程序运行 chrome,postgresql）
22. ~~限制 ssh 用户命令 restricted-ssh-commands~~
23. Haveged 增强随机数,~~rngd.~~
24. 禁止控制台 ctrl alt del 重启
25. 开启内核 audit 审计，审计规则：修改关键文件,修改标志位,修改能力,修改扩展属性
26. 系统安全审计工具 lynis
27. Dmesg 限制，普通用户无法查看 dmesg
28. Pam_tmpdir 实现用户使用独立的 tmp 目录
29. 禁止普通用户查看日志（尽可能多的禁止日志，直至最终完全不可见）
30. 禁止普通用户（非 adm 组成员）查看 last 登录信息
31. 用户进程资源限制 Pam limit（避免 fork bomb 威胁,以及各种资源耗尽导致服务器宕机）
32. ~~Fail2ban 根据日志执行相关屏蔽动作~~
33. ~~防火墙软件 nftables（替代 iptables）~~
34. ~~Suricata 入侵检测（替代 snort）~~
35. ~~蜜罐软件 honeyd，farpd，iisemulator，tinyhoneypot~~
36. 系统完整性检查工具 aide，~~AFICK，Osiris，Samhain，Tripwire，OSSEC(syscheck)，mtree~~
37. Rootkit 检查工具 unhide，~~Rkhunter~~
38. ~~普通用户隐藏硬件信息（使用 grsecurity 的 RBAC 禁止查看硬件命令，禁止访问 procfs 和 sysfs 实现）~~

39. 缓冲区溢出检查工具 pscan, flawfinder, splint, bfbtester

40. 记录 unix 用户创建、删除

41. 用户命令审计追踪 snoopy,日志/var/log/auth

42. IMA（Integrity Measurement Architecture 完整性度量体系）

43. EVM（Extended Verification Module 扩展验证模块）

44. TPM（可信平台模块，需要硬件支持，当前不支持 TPM2.0）tpm-tools, trousers, tss2

45. 加密工具 gnupg, ecryptfs-utils, cryptsetup,

46. Neopi （Webshell 代码检测）

47. psad,（根据 iptables 日志探测端口扫描），sxid （检查设置 suid, sgid 的文件和目录）

48. DNS 解析审计

49. 防火墙使用全球 IP 地址实时黑名单

50. Nginx+modsecurity WAF 防火墙

51. 隐藏系统内关键进程，秘密实施审计、监视、记录系统行为(由 grsecurity+RBAC 实现)

52. apparmor 基于路径的强制访问控制（3.0 支持网络访问控制，支持 grsecurity 内核，但是不能与 grsecurity 的 RBAC 同时开启）

参考：

中国国家标准 GB/T 18336-2008

ISO 国际标准 ISO/IEC 18045:2005

美国国防部：DISA STIG 标准，git clone https://github.com/hardenedlinux/STIG-4-Debian.git

FLOSS 最佳实践

https://wiki.archlinux.org/index.php/Security

https://www.owasp.org/index.php/PHP_Security_Cheat_Sheet#Weak_typing

https://github.com/linuxfoundation/cii-best-practices-badge/blob/master/doc/criteria.md

https://trent.utfs.org/wiki/Hardening/Linux#Per-user_.2Ftmp

http://netfilter.org/projects/nftables/

https://www.debian.org/doc/manuals/securing-debian-howto/index.en.html

http://www.honeynet.org/

http://wiki.debian.org/SELinux

http://www.openwall.com/linux/

https://linux-audit.com/nftables-beginners-guide-to-traffic-filtering/

http://www.malwaremustdie.org/

http://iase.disa.mil/stigs/Pages/index.aspx

http://www.astra-linux.com 俄罗斯军队的 GNU/Linux 发行版

https://www.rsbac.org/

https://wiki.gentoo.org/wiki/Integrity_Measurement_Architecture

https://lwn.net/Articles/394170/

https://sourceforge.net/p/linux-ima/wiki/Home/

http://ecryptfs.org/documentation.html

https://www.badips.com

http://www.modsecurity.org/

https://www.owasp.org

http://flint.cs.yale.edu/certikos/ CertiKOS

https://wiki.gentoo.org/wiki/Hardened_Kernel

https://github.com/thestinger/linux-hardened
https://github.com/bartblaze/Disable-Intel-AMT

注释
被删除线划掉的功能特性，仅仅是默认的初始安装未实现，可以根据实际情况启用。