Teste de Sistemas Operativos

Correcção do Grupo I

fsm

Junho 2017

Resumo

Pretende-se com este documento ajudar a compreender a forma correcta de responder ao Grupo I do teste de Sistemas Operativos (SO) de 26 de Maio de 2016 (adiado para 6 de Junho), permitindo uma melhor preparação para os alunos que terão de comparecer ao exame de recurso. A intenção não é dar aqui soluções completas, "replicáveis" às cegas em qualquer outra prova de avaliação. A ideia é ficar a perceber os caminhos errados, quer de interpretação de enunciado quer de estratégia de resposta. Quando chegar ao fim do texto deverá ser capaz de, por si só, chegar à resposta correcta.

1 Antes de começar...

As provas de avaliação destinam-se a mostrar os resultados da aprendizagem. Aprendizagem. Passar a ompreender novos problemas, equacionar soluções, optar e saber justificar essa escolha, implementar ou descrever a solução, conforme o caso, e finalmente demonstrar que funciona no caso concreto que foi apresentado. Em SO reproduzir um "padrão" vale muito pouco. Não é o resultado ("output") que interessa, mesmo que possa estar correcto. Primeiro há que provar que é trabalho / aprendizagem pessoal e ser-se capaz de descrever o que faz e que benefício traz este algoritmo, esta estratégia, o uso desta arquitectura ou system call, etc.

A palavra de ordem é justificar. Respostas descritivas, sem justificação e compromisso pessoal são equivalentes à antiga "morte súbita" do futebol: acabou o jogo por aqui. A resposta deve ser sucinta, as tais 10 a 15 linhas (para obrigar a pensar e selecionar o que realmente interessa) usando com precisão as palavras que têm significado especial em SO.

2 Primeira Questão

2.1 Enunciado

Praticamente todos os dias surgem notícias de ataques a sistemas informáticos e da necessidade de protecção. Ora sendo o sistema operativo um gestor de recursos, também lhe cabe a tarefa de fornecer mecanismos para proteger esses recursos. Escolha três recursos estudados nas aulas teóricas e explique os respectivos mecanismos de protecção. Assuma que se trata de um sistema com Linux.

Infelizmente é verdade que todos dias temos notícias de ataques informáticos, de muitas formas, e que há uma parafernália de tentativas de defesa, desde *firewalls*, anti-vírus, sistemas de backup, e por aí adiante. No entanto, convém começar por colocar esta questão no contexto da unidade curricular de SO.

Para nós, falar do sistema operativo significa falar do *kernel*, aquela camada inicial que, numa linguagem muito informal, "vai até às *system-calls*". Não são as aplicações que são instaladas juntamente com o Linux ou descarregadas da internet, nem são as muitas janelas que se podem abrir no monitor. É o kernel e os recursos aí geridos. Além disso, em SO tudo se passa dentro de uma só máquina; assuntos de "redes" e "sistemas distribuídos" são outras águas. Os dois primeiros exemplos dados acima (há quem lhes chame erradamente "recursos") não fazem parte da matéria de SO, e o terceiro é pouco provável que tenha algo a ver com a pergunta. Esta refere especificamente defesa (pelo kernel) de recursos contra ataques informáticos. Backup é uma tarefa de administração, que nos computadores pessoais deve ser feito por quem os utiliza e nas máquinas partilhadas ou servidores será feito "centralmente" por administradores ou operadores de sistemas.

Embora o backup seja recomendado "por razões de segurança", não é o kernel do SO que toma a iniciativa de o fazer. E, para os mais distraídos, o próprio backup pode também ser atacado e adulterado.

O que é então um ataque informático? Sem complicar, dando exemplos muito rebuscados (que os há, infelizmente). Vamos a exemplos concretos: e-mails, fotografias, exames (médicos ou académicos), processos judiciais, enfim "escândalos" empolados pelos media mas onde se encontra algo de comum. Já descobriu? Então vamos a mais exemplos: mudar a nota de um teste, não nos Serviços Académicos porque aí seria numa base de dados e essas não são matéria nossa (mais ou menos...). Refiro-me a (tentar) mudar entrando na máquina de um professor, isso sim já é SO[1]... Ou os tão falados ataques de Ransomware em que a informação é cifrada e é pedido um resgate em bitcoins.

Que há de comum em todos esses ataques? É simples: são exemplos em que a informação existente na máquina em causa foi vista (e eventualmente divulgada) por quem não tinha direito de o fazer, ou foi deliberadamente modificada.

Todos sabemos que a "informação" é gerida pelo sistema de ficheiros, um dos grandes componentes de um sistema operativo. Ficheiros, directorias (e também periféricos e FIFOS no caso do Unix e seus descendentes) possuem um conjunto de permissões que, *entre outras*, permitem restringir o acesso para leitura, escrita e execução a 3 classes de utilizadores: dono, grupo a que pertence e finalmente os outros utilizadores. Infelizmente para a esmagadora maioria dos alunos que responderam com muita pressa a esta pergunta, as permissões são só metade da resposta. São as "fechaduras"... e é preciso ter a chave certa para abrir as respectivas portas. Onde estão essas chaves? Antes disso, em que momento se validam as permissões?

Não responda sem pensar que é na operação de leitura ou escrita... Primeiro porque a leitura ou escrita se realiza com as system calls *read* e *write* e essas só permitem acesso se antes tiver passado pelo *open* sem erros. É no *open* que se fazem as validações. Depois porque a validação da execução é feita no *exec*, que na prática também vai ter necessidade de "abrir" o ficheiro executável. Sendo então na abertura, como é que dentro da system call *open* se determina que o programa pode aceder a esse recurso? Esta é fácil, qualquer programa foi colocado em execução por um determinado utilizador, e por isso basta verificar as permissões do utilizador "corrente", associado ao programa em execução. E como se faz isso? Agora sim, entramos na segunda parte da resposta, a que irá garantir 100% da cotação.

Cada objecto com entrada no sistema de ficheiros (pense em ficheiro ou directoria) tem a indicação do dono e basta comparar se o utilizador corrente é dono desse objecto, está no grupo ou pertence aos "outros". Isso é em português, é vago... Em SO é preciso dizer que em Linux cada utilizador é identificado por um número, o seu USER ID, que é registado pelo sistema quando esse utilizador é criado. Onde? Pode ser em vários locais mas para efeito desta pergunta de teste bastaria referir o ficheiro de passwords. E isso leva-nos à pergunta quase final: como é que se sabe que o programa que está a tentar aceder a um recurso que foi criado pelo utilizador 501 ou outro qualquer é mesmo o dono, ou elemento de um grupo com acesso permitido? Resposta: autentica-se o utilizador. E como se autentica? Pede-se a password. Quando? No momento do login[2]. Era preciso dizer isso tudo? Sim, mas bastava uma frase.

Em resumo, para ter a resposta certa teria de, em poucas linhas, falar em permissões e na autenticação de quem quer aceder. Tal como num banco ou num exame, o cheque ou folha de respostas só tem valor se o autor(a) for autenticado. E, sim, os ficheiros de passwords e grupos são de acesso restrito a quem administra a máquina, que também terá de se autenticar e, tal como todos nós, terá de zelar para que a sua password se mantenha pessoal. Por isso muitos dos ataques resultam de passwords fracas ou de "engenharia social", informação pública ou que os utilizadores ingenuamente disponibilizam (através de técnicas de *sniffing* ou simplemente indo buscar inspiração às redes sociais ou a um simples "gosto tanto do meu cão"). No caso do ransomware os ataques são normalmente mais complicados e exploram vulnerabilidades que vieram a público e não foram atempadamente corrigidas. No entanto, em última análise, conseguem executar operações com permissões que não deveriam ter.

[1] Convém acrescentar que já há muitos anos que entrar em computador alheio é crime, há leis que punem esses comportamentos.

[2] A validação do grupo implica consulta do ficheiro que lista todos os grupos a que pertence cada um dos utilizadores. No entanto, já não seria necessário falar disso no teste.