Ambientes de Execução Seguros
2º Semestre, 2020/21

1º Exame
7 de julho de 2021

- Todas as perguntas têm a mesma cotação.
- A duração total do exame é de 3 horas.

1. Um dos mecanismos fundamentais de uma *Trusted Computing Base* é a existência de mecanismos de hardware e software que permitem concretizar um monitor de controlos de acesso, o qual terá como reponsabilidade a proteção de recursos partilhados do sistema. Explique como é que os sistemas operativos (software) exploram funcionalidades do hardware, nomeadamente do CPU, para concretizar este monitor de controlo de acesso.

2. Considere o conceito de atestação remota, realizada através de um TPM. Indique, genericamente, o que se consegue provar com a mesma.

3. Explique por que razão um arranque seguro (*secure bootstrap*) é vital para evitar potenciais corrupções da *Trusted Computing Base* concretizada por um sistemas operativo.

4. Os processos em Linux possuem um *effective* UID e um *real* UID, e o mesmo para o seu *default* GID. Explique por que razão os processos têm estes dois UID.

5. Considere o mecanismo Set-UID do Linux. Explique como é que mesmo funciona.

6. O suporte para a virtualização diretamente sobre o hardware (*bare-metal*), e não sobre um sistema operativo (*hosted*), obrigou os fabricantes de CPUs a introduzirem novos níveis de proteção. Explique quais e porquê.

7. Explique como funcionava o modelo de *secure boot* proposto no âmbito do AEGIS (não precisa de referir o processo de recuperação).

8. O TPM providencia um conjunto de registos, denominados por PCR (*Platform Configuration Register*). Explique como funcionam estes registos.

9. Explique como é que os PCR podem ser usados para demonstrar que um processo de *secure boot* foi efetivamente concretizado.

10. Uma chave de cifra do sistema de ficheiros de um sistema operativo (e.g. a usada pelo Microsoft BitLocker) pode estar protegida por um TPM com base numa atestação feita com um determinado conjunto de PCR (*Platform Configuration Register*), por exemplo, o que monitoriza o *bootloader*. Que garantias são dadas, nesse caso.

11. Segundo as definições do TCPA (*Trusted Computing Platform Alliance*), qual é a diferença entre um *authenticated boot* e um *secure boot*?

12. Como é realizada a atestação remota de um processo de arranque (*bootstrap*) usando o TPM?

13. Para que serve e como funciona o UEFI *secure boot*?

14. Um enclave SGX protege a execução de código da observação por outras tarefas que executem no mesmo processador, independentemente do seu nível de privilégio (*protection ring*). Explique como.

15. A arquitetura ARM TrustZone permite executar sobre o mesmo CPU (ARM) dois sistemas isolados: *Rich OS* (não seguro) e *Secure OS* (seguro). Explique de que forma os dois interagem entre si.

16. O TPM possui 4 hierarquias: *platform*, *storage*, *endorsement* e NULL. Explique para que serve cada uma.

17. Cada hierarquia do TPM possui uma árvore de chaves, as quais começam em chaves primárias. Explique o processo de criação destas chaves.

18. Um TPM tem a capacidade de exportar dados gerados dentro de si de tal forma que, mais tarde, consegue verificar que foram por si gerados aquando de uma futura importação. Explique como.

19. As chaves primárias da hierarquia *endorsment* do TPM são normalmente certificadas pelo fabricante do respetivo TPM. Explique porquê.

20. Um *namespace* Linux pemite definir universos onde alguns recursos do sistema operativo são observados de uma forma condicionada. Considerando o caso no *namespace* de processos, ou seja, dos seus identificadores (PID), explique que condicionalismo é realizado.