Principles of Computer Security

Alvin Lin

January 2018 - May 2018

Exam 2 Key

1. The main issue is with the `gets()` function, as there is no check and overflows the variable buf. Some ways to fix the issue: use `fgets()`, or read a fixed number of characters, not an unchecked read.

2. The canary is generated using random numbers to make it harder for attackers to guess the canary value.

3. No preemption can lead to deadlock. Here, the system is preventing "no preemption" to prevent deadlocks. Since the system will allow preemption of resources, a programmer's abilities to write secure code is impacted because preemption will occur, and will need expert handling. Exception handling will need to become the default.

4. Exponential back-off means retrying when a deadlock is reported after waiting for $2n$ seconds before retrying until some value n such as 10 is reached, and then aborting the program. If a DDOS attack is occurring via deadlocks, then this technique is ineffective and actually gets in the way by delaying mitigation.

5. A pseudo-random number generator allows an attacker to make 100% accurate guesses, thus allowing the attacker to generate receipt URLs with 100% success. Don't use pseudo-random number generator, or use a different technique to store receipts that is backed by authentication.

6. This is a classic TOCTOU attack discussed in class. The file checked on lines 1-2 may not be the one used in line 8 because the race condition caused by the script could have changed the file. The check for file permission should be combined with the file open in an atomic operation.

7. `grant select(name) on Student to ‘abc1234’`
   SQL does not provide row level security. To do this, we would need to create a view for that student:
   `create view AbcView as select * from Student where name=‘abc1234’`

8. As roles in the database cannot be used in the middleware, it means that the system must implement its own RBAC in the middleware. The use of RBAC in the database cannot be relied upon, and the system will need to maintain two sets of RBAC.

9.

Unobservability	User may use resource or service without others being aware of its use.
Anonymity	Use resource or service without disclosing their identity.
Unlinkability	Use multiple resources or services without others being able to link these uses together.
Pseudonymity	Use a resource or service without disclosing their identity, but can still be accountable for that use.

10. Both have same or similar functionality. Data perturbation needs a copy of the entire dataset so space performance is poor. It has good runtime performance as data has already been perturbed so time performance is good. Output perturbation does not need a copy of the entire dataset so space performance is good. Output perturbation has poor runtime performance however, since data has to be perturbed on the fly.

11. A user can enter “123 or true –” to make the WHERE clause always evaluate to true, which would cause the database to return every account. Users can inject arbitrary strings into the query and nullify the rest of the query using the SQL comment. The best way to defend from this would be to use input sanitization and parameterized replacement when generating the query string.

12. SQL does not provide row level security.

You can find all my notes at `http://omgimanerd.tech/notes`. If you have any questions, comments, or concerns, please contact me at alvin@omgimanerd.tech