

CE4062/CZ4062

Computer Security

Tutorial 5: Operating System Security

Tianwei Zhang

Q1

Short answers

- (a) Give an example of how a rootkit can compromise the system after obtaining the root privilege.
- (b) Briefly describe three stages employed by the OS
- (c) What is the controlled invocation? What is its potential danger?

Solution

Short answers

- (a) Give an example of how a rootkit can compromise the system after obtaining the root privilege.

A rootkit can compromise different kernel structures to achieve malicious behaviors. For instance,

- ▶ *It can change a function pointer in the system call table to make it point to a malicious function.*
- ▶ *It can directly change the system call function, making it jump to a malicious function.*
- ▶ *It can change a function pointer in the interrupt descriptor table to make it point to a malicious function.*

Solution

Short answers

- b) Briefly describe three stages employed by the OS
 - ▶ *Identification & Authentication: authenticate if a user attempting to enter the system is correct or not.*
 - ▶ *Access Control: when a subject (process, user, ...) wants to access an object (file, network socket, ...), check if such access is allowed.*
 - ▶ *Logging & auditing: record all protection-orientated activities, important to understand what happened, why, and catching things that shouldn't*

Solution

Short answers

- c) What is the controlled invocation? What is its potential danger?

A user executes a privileged program by inheriting the permission of the program's owner. This is usually achieved by enabling SUID

As the user has the program owner's privileges when running a SUID program, the program should only do what the owner intended. By tricking a SUID program owned by root to do unintended things, an attacker can act as the root.

Q2

Consider a computer system with three users: Alice, Bob, and Cindy. Alice owns the file *alicerc*, and Bob and Cindy can read it. Cindy can read and write the file *bobrc*, which Bob owns, but Alice can only read it. Only Cindy can read and write the file *cindyrc*, which she owns. Assume that the owner of each of these files can execute it.

- (a) Create the corresponding access control matrix.
- (b) Cindy gives Alice permission to read *cindyrc*, and Bob removes Alice's ability to read *bobrc*. Show the new access control matrix
- (c) Show the capabilities associated with Alice.

Solution

Consider a computer system with three users: Alice, Bob, and Cindy. Alice owns the file *alicerc*, and Bob and Cindy can read it. Cindy can read and write the file *bobrc*, which Bob owns, but Alice can only read it. Only Cindy can read and write the file *cindycrc*, which she owns. Assume that the owner of each of these files can execute it.


- a. Create the corresponding access control matrix.

	alicerc	bobrc	cindycrc
Alice	Execute	Read	
Bob	Read	Execute	
Cindy	Read	Read Write	Read Write Execute

Solution

Consider a computer system with three users: Alice, Bob, and Cindy. Alice owns the file *alicerc*, and Bob and Cindy can read it. Cindy can read and write the file *bobrc*, which Bob owns, but Alice can only read it. Only Cindy can read and write the file *cindyrc*, which she owns. Assume that the owner of each of these files can execute it.

- b. Cindy gives Alice permission to read *cindyrc*, and Bob removes Alice's ability to read *bobrc*. Show the new access control matrix.



	alicerc	bobrc	cindyrc
Alice	Execute	Read	Read
Bob	Read	Execute	
Cindy	Read	Read Write	Read Write Execute

Solution

Consider a computer system with three users: Alice, Bob, and Cindy. Alice owns the file *alicerc*, and Bob and Cindy can read it. Cindy can read and write the file *bobrc*, which Bob owns, but Alice can only read it. Only Cindy can read and write the file *cindyrc*, which she owns. Assume that the owner of each of these files can execute it.

- c. Show the capabilities associated with Alice.

	alicerc	bobrc	cindyrc
Alice	Execute		Read
Bob	Read	Execute	
Cindy	Read	Read Write	Read Write Execute

{alicerc: {execute}; bobrc: {}; cindyrc: {read}}

Q3

Let's consider the scenario in Q2 again. Assume this is the Unix system. The users Bob and Cindy are in the same group, while Alice is in a different group.

- (a) For the original access control matrix in Q2(a), please write the permission for the files *alicerc*, *bobrc* and *cindyc*.
- (b) To adjust the permissions in Q2(b), please write the corresponding commands for *cindyc* and *bobrc*, respectively.

Solution

Let's consider the scenario in Q2 again. Assume this is the Unix system. The users Bob and Cindy are in the same group, while Alice is in a different group.

- a. For the original access control matrix in Q2(a), please write the permission for the files *alicerc*, *bobrc* and *cindyrc*.

alicerc: --x---r-- 104
bobrc: --xrw-r-- 164
cindyrc: rwx----- 700

	alicerc	bobrc	cindyrc
Alice	Execute	Read	
Bob	Read	Execute	
Cindy	Read	Read Write	Read Write Execute

Solution

Let's consider the scenario in Q2 again. Assume this is the Unix system. The users Bob and Cindy are in the same group, while Alice is in a different group.

- b. To adjust the permissions in Q2(b), please write the corresponding commands for *cindyc* and *bobrc*, respectively.

alicerc: --x---r--

bobrc: --xrw----

cindyc: rwx---**r**--

chmod 160 bobrc

chmod 704 cindyc

chmod o-r bobrc

chmod o+r cindyc

	alicerc	bobrc	cindyc
Alice	Execute	Read	← Read
Bob	Read	Execute	
Cindy	Read	Read Write	Read Write Execute

Q4

A group of researchers is working on analyzing web search results from a major Internet search provider. At the Internet company, a group of search engineers collects and updates databases of: search queries, IP (internet protocol) addresses where the queries came from and timestamps for the queries made by online users. A search manager is in charge of the group of engineers, and can read the query and timestamps database, but not the IP address database -- due to privacy concerns. The researchers are able to access the databases with read-only privileges. The general public should not have access to the database for privacy reasons.

- (a) Complete the access control matrix by filling in the access permissions for the different objects shown. Each entry can be either read, write, read/write, or '-' (for no access).
- (b) List the ACLs for each object (or class of objects)
- (c) List the capabilities for each subject (or class of subjects)

Solution

A group of researchers is working on analyzing web search results from a major Internet search provider. At the Internet company, a group of search engineers collects and updates databases of: search queries, IP (internet protocol) addresses where the queries came from and timestamps for the queries made by online users. A search manager is in charge of the group of engineers, and can read the query and timestamps database, but not the IP address database -- due to privacy concerns. The researchers are able to access the databases with read-only privileges. The general public should not have access to the database for privacy reasons.

- a. Complete the access control matrix by filling in the access permissions for the different objects shown. Each entry can be either read, write, read/write, or '-' (for no access).

	Query	IP	Timestamp
Search Engineers	rw	rw	rw
Search Manager	r	-	r
Researchers	r	r	r
Public	-	-	-

Solution

- b. List the ACLs for each object (or class of objects)

Query: {SE: {rw}; SM: {r}; R: {r}; P: {-}}

IP: {SE: {rw}; SM: {-}; R: {r}; P: {-}}

Timestamp: {SE: {rw}; SM: {r}; R: {r}; P: {-}}

	Query	IP	Timestamp
Search Engineers	rw	rw	rw
Search Manager	r	-	r
Researchers	r	r	r
Public	-	-	-

Solution

- c. List the capabilities for each subject (or class of subjects)

Search Engineers: {Query: {rw}; IP: {rw}; Timestamp: {rw}}

Search Manager: {Query: {r}; IP: {-}; Timestamp: {r}}

Research: {Query: {r}; IP: {r}; Timestamp: {r}}

Pub lic: {Query: {-}; IP: {-}; Timestamp: {-}}

	Query	IP	Timestamp
Search Engineers	rw	rw	rw
Search Manager	r	-	r
Researchers	r	r	r
Public	-	-	-