SIM GLOBAL EDUCATION

UNIVERSITY OF WOLLONGONG AUSTRALIA

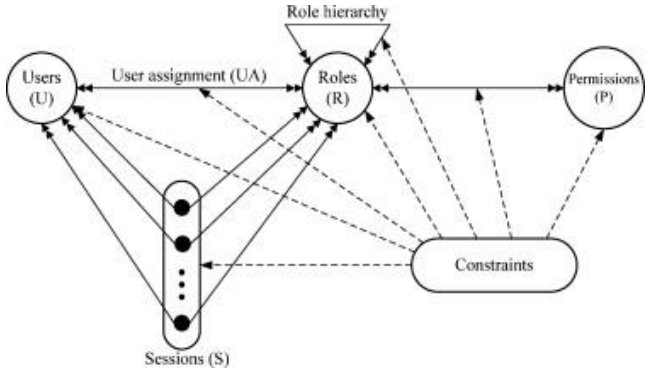
CSCI262 – SYSTEM SECURITY

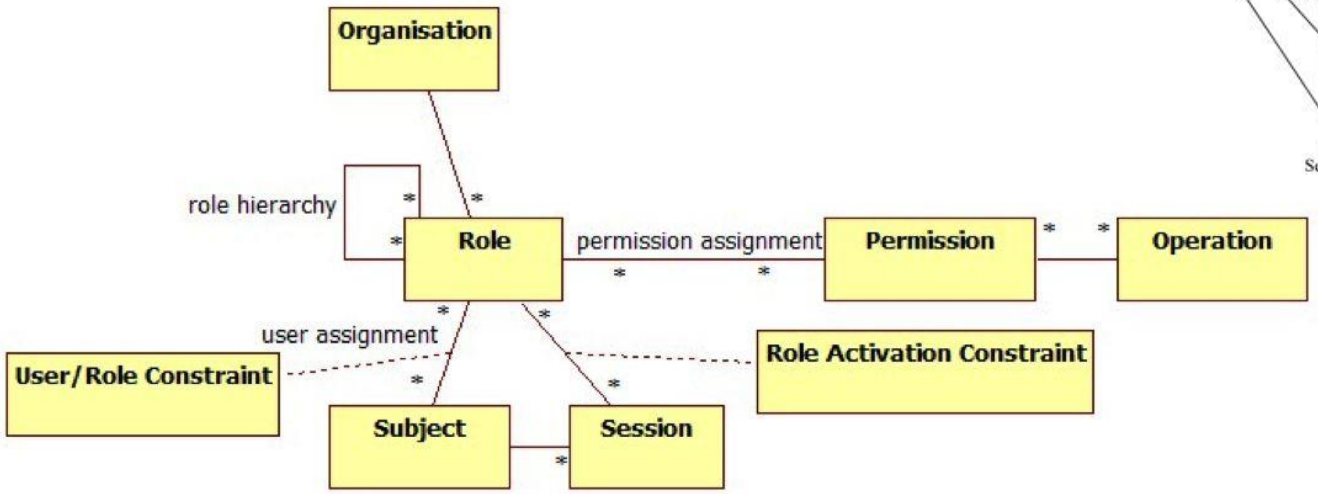
Tutorial Set C

5 November 2022

Tutorial Set C - Part One

1. Explain RBAC.jpg (taken from Wikipedia) and Figure 1 in SandhuET1996.pdf, in the context of Role Based Access Control. In particular, explain what is meant by the constraints and role hierarchies.





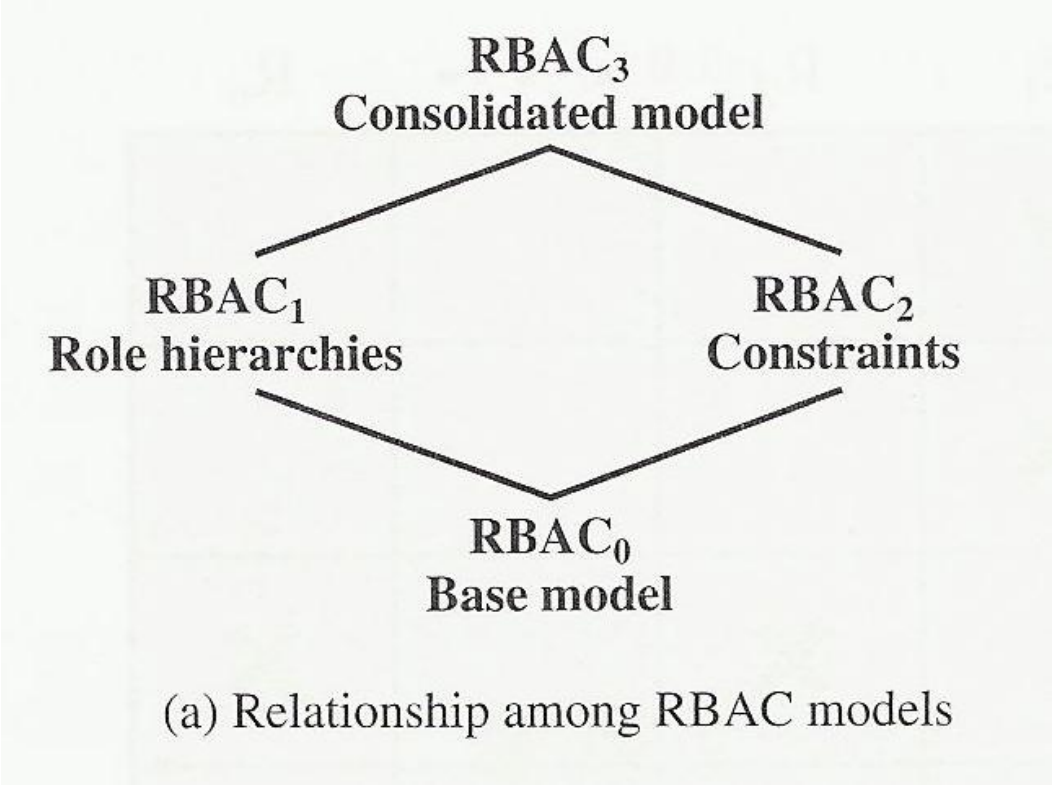
Tutorial Set C - Part One

- Role-based Access Control (RBAC) defines the access rights of individuals based on the roles that users assume in a system.
- Typically, a role is defined as a job function within an organization, for example, A director.
- Access rights are assigned to roles, and in turn, users are assigned to different roles, either statically or dynamically.
- RBAC is flexible; it can implement mandatory access control (MAC) or discretionary access control (DAC).

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

Tutorial Set C - Part One

- For the ongoing standardization efforts, the RBAC model is defined using four conceptual model as shown below.



(a) Relationship among RBAC models

Tutorial Set C - Part One

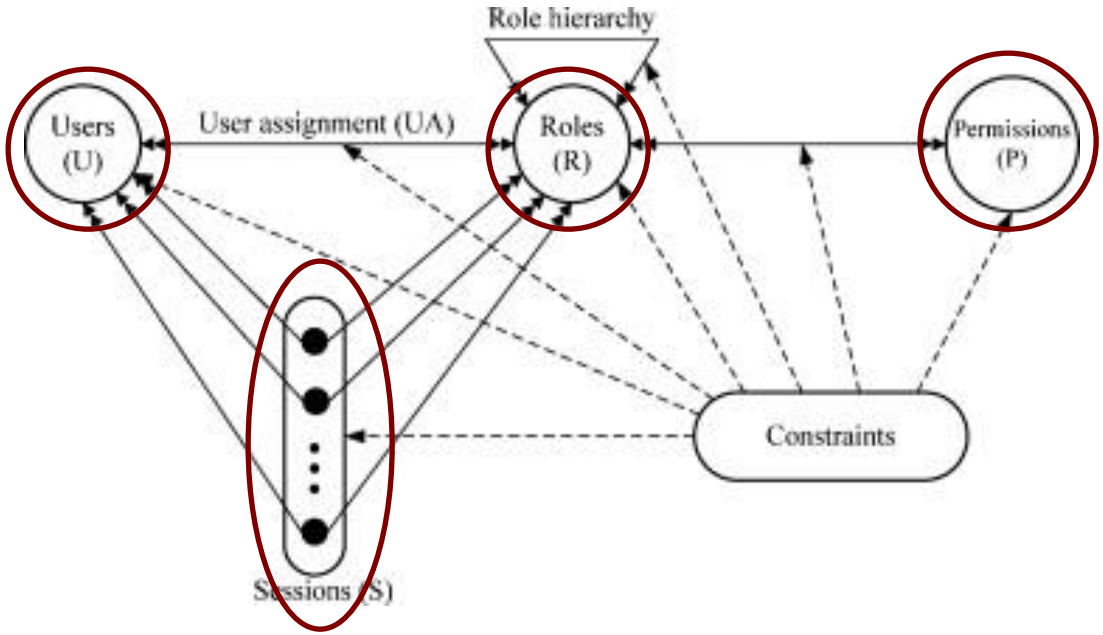
Components of RBAC base Model ($RBAC_0$):

- **User**: An individual that has access to a computer system. Each individual has an associated user ID.
- **Role**: A named job function within an organization that controls a computer system. Typically, associated with each role is a description of the authority and responsibility conferred on this role, and on any user who assumes this role.
- **Permission**: An approval of a particular mode of access to one or more objects. Equivalent terms are *access right*, *privilege*, and *authorization*.

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

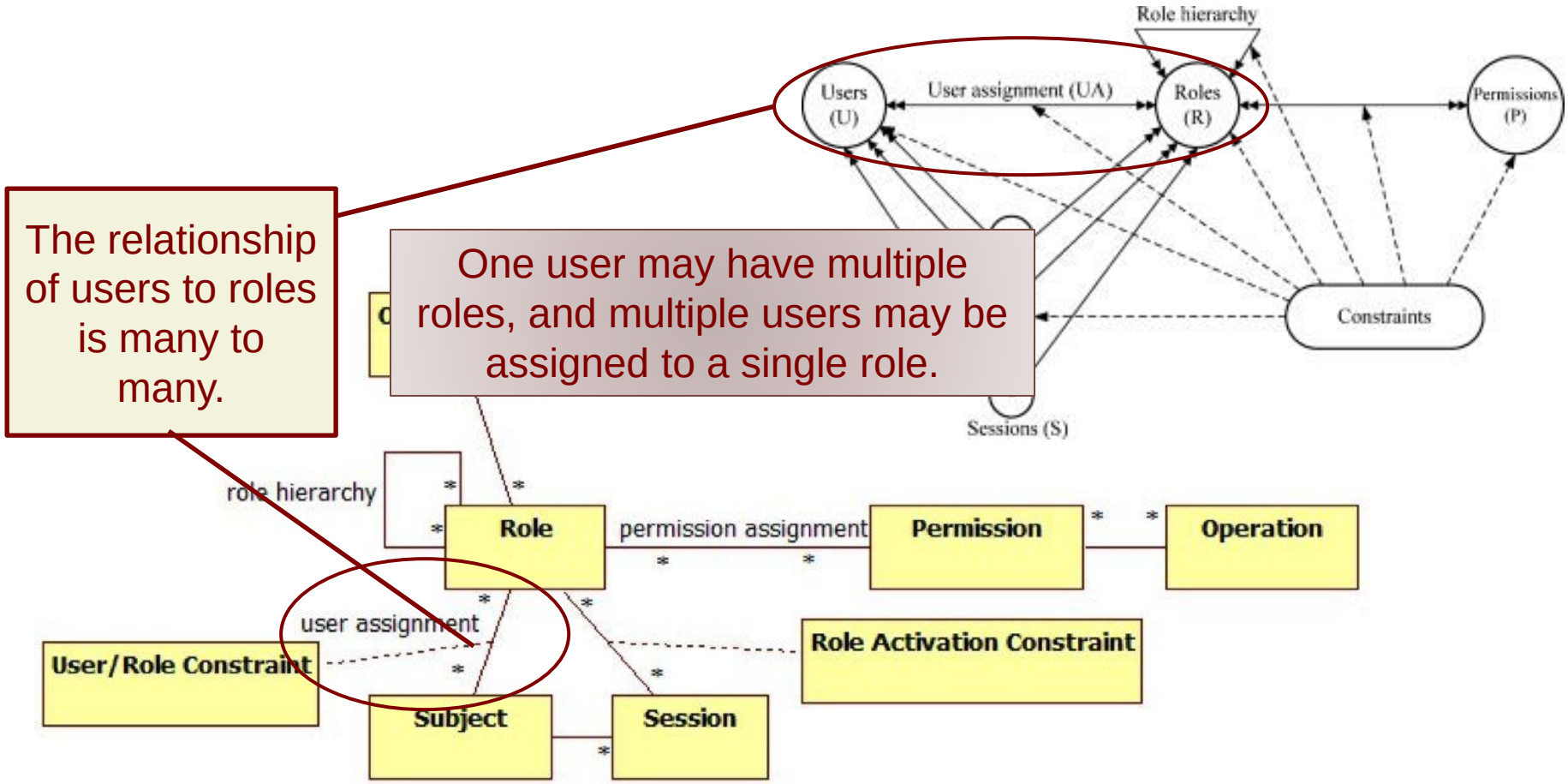
Tutorial Set C - Part One

- **Session:** a mapping between a user and an activated subset of the set of roles to which the user is assigned.



William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

Tutorial Set C - Part One



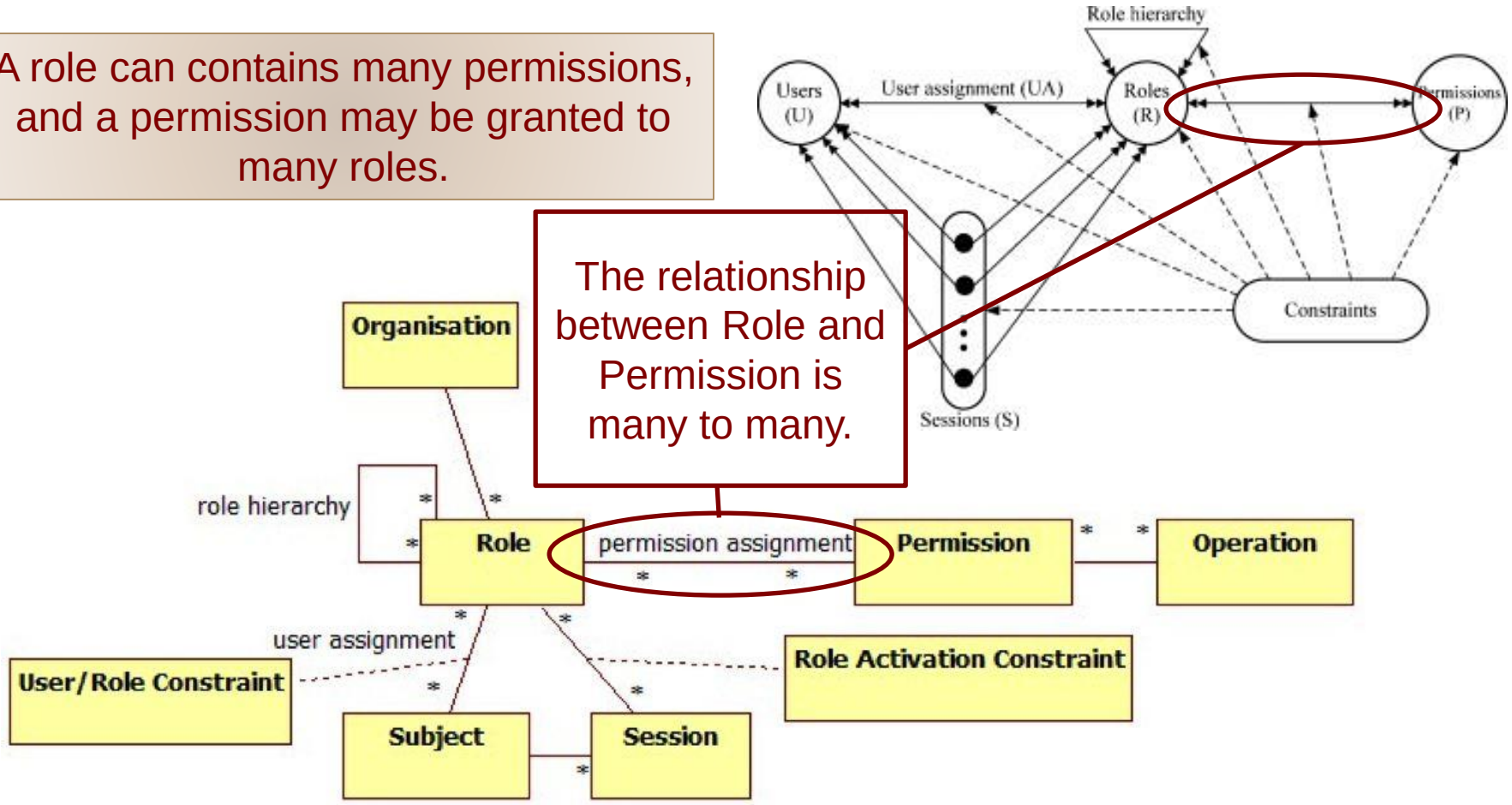
The relationship of users to roles is many to many.

One user may have multiple roles, and multiple users may be assigned to a single role.

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

Tutorial Set C - Part One

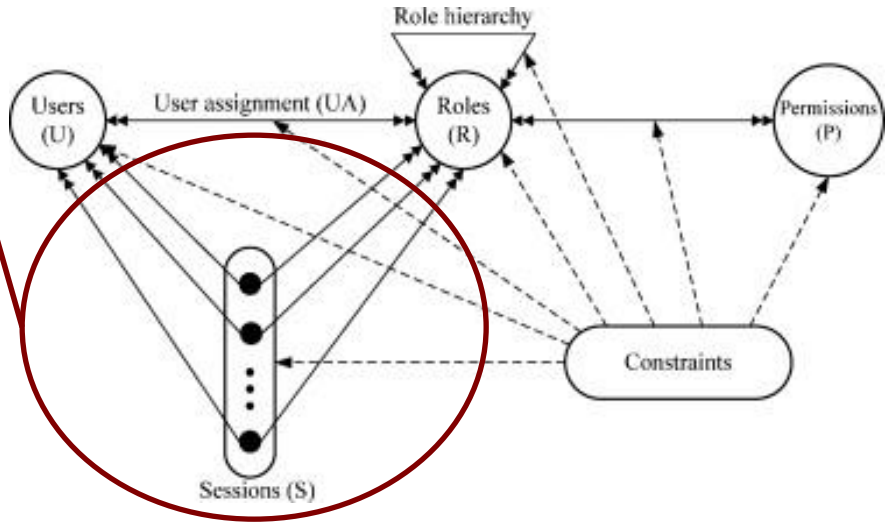
A role can contains many permissions, and a permission may be granted to many roles.

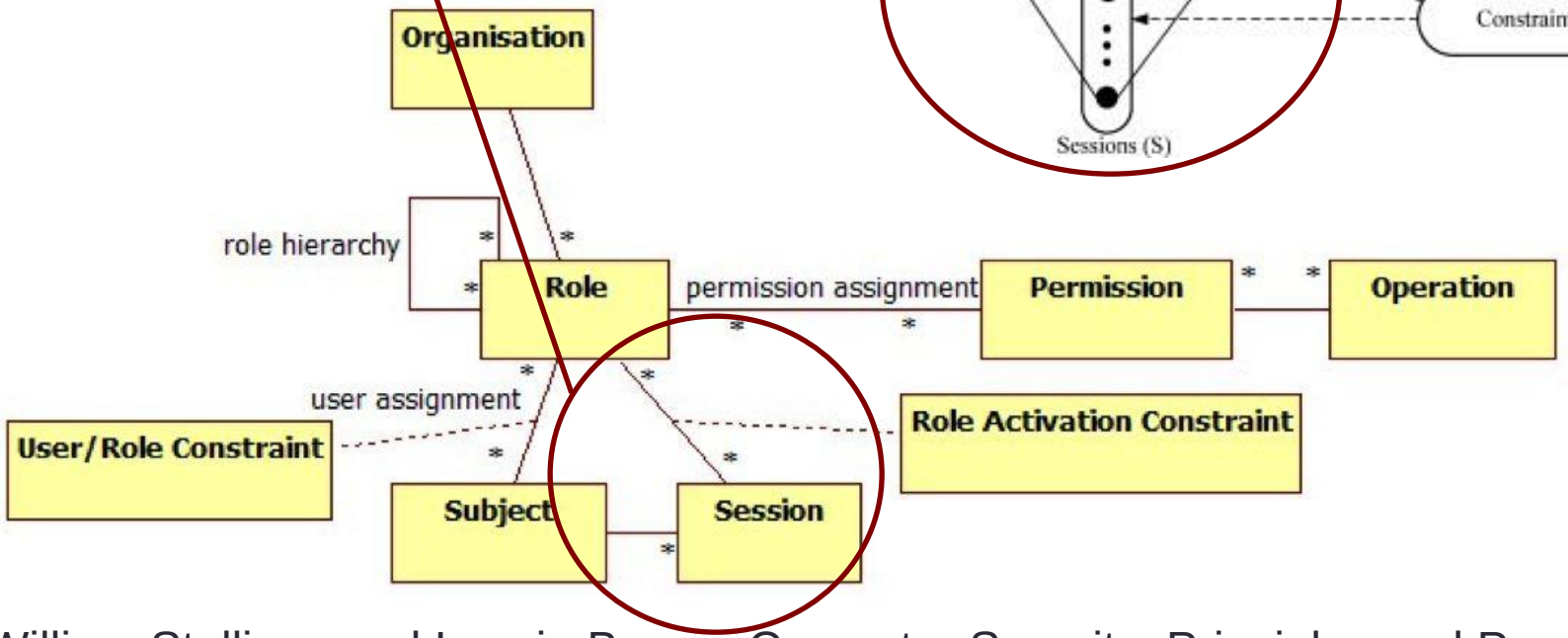


William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

Tutorial Set C - Part One

A session is an association between a user and one or more roles to which the user has been assigned.





William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

Tutorial Set C - Part One

Role Hierarchies Model ($RBAC_1$):

- This model provides a means of reflecting the hierarchical structure of roles in an organization.

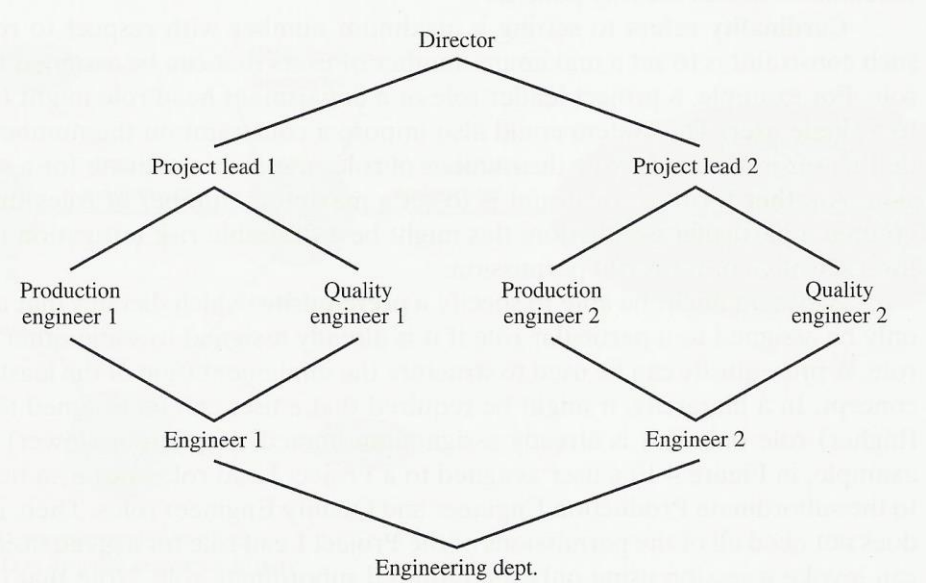


Figure 4.10 **Example of Role Hierarchy**

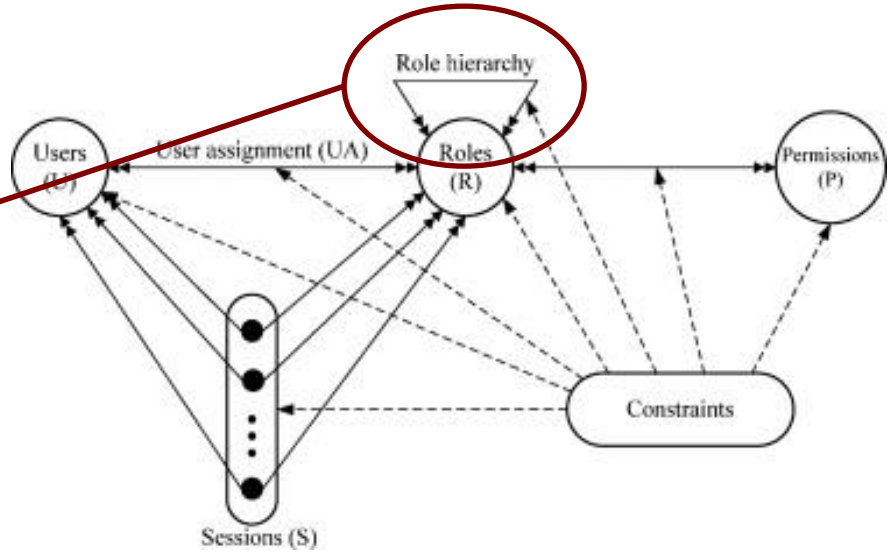
Typically, job functions with greater responsibility have greater authority to access resources.

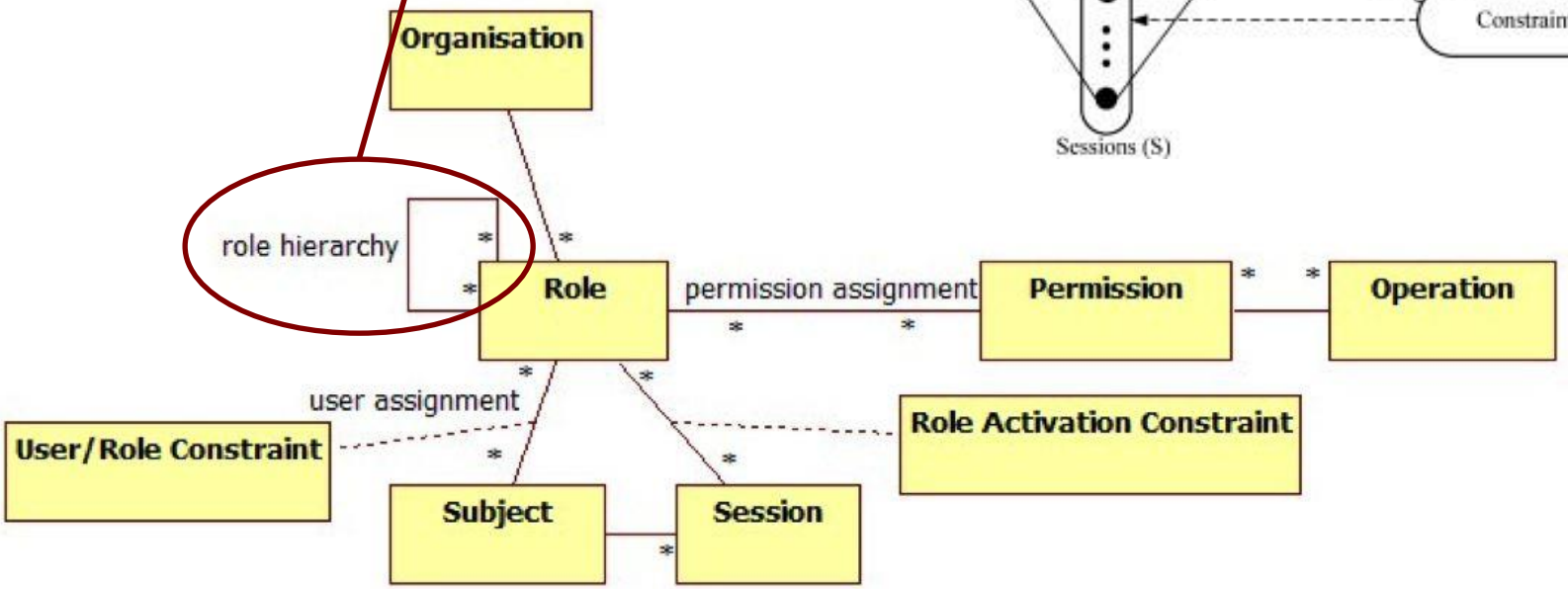
A subordinate job function may have a subset of the access rights of the superior job function.

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

Tutorial Set C - Part One

Role hierarchies make use of the concept of inheritance to enable one role to implicitly include access rights associated with a subordinate role.





William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

Tutorial Set C - Part One

Constraints Model ($RBAC_2$):

- This model provides a means of adapting RBAC to the specifics of administrative and security policies in an organization.
- Three types of constraints proposed and implemented:
  - Mutually exclusive roles
    - The same user can be assigned to at most one role in a mutually exclusive set. Mutually exclusive constraint supports a separation of duties and capabilities within an organization.

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

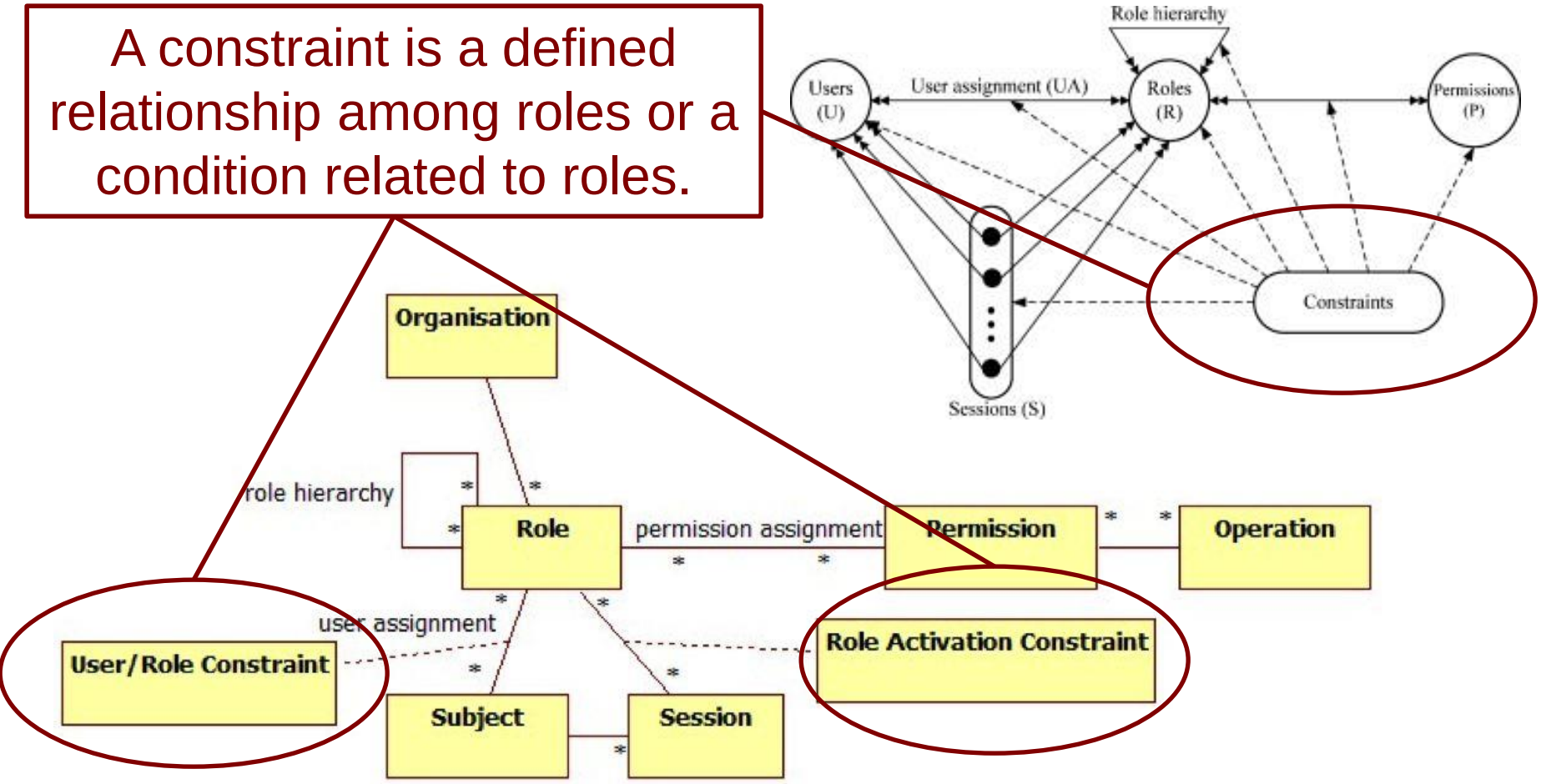
Tutorial Set C - Part One

- Cardinality
  - Cardinality refers to setting a maximum number with respect to roles.
  - One such constraint is to set a maximum number of users that can be assigned to a given role. For example, only one person can fill the role of department chair.
- Prerequisite
  - Prerequisite dictates that a user can only be assigned to a particular role if it is already assigned to some other specified role.

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

Tutorial Set C - Part One

A constraint is a defined relationship among roles or a condition related to roles.



William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012, pp 143 - 151.

Tutorial Set C - Part One

Consolidated model ($RBAC_3$):

- Consolidated model provides both role hierarchies and constraints, as it combines $RBAC_1$ and $RBAC_2$.

Tutorial Set C - Part One

2. What is a zombie in the context of denial of service?

Zombie, also known as bot, is a malware (malicious software) that secretly takes control over a computer that is connected to the Internet, and use it to launch attack. With zombie, it is hard to trace to the attacker.

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2008, page 240.

Tutorial Set C - Part One

3. What is the default size of a ping?

   A ping is normally 56 bytes in size; 84 bytes if the Internet Protocol header is considered.

Tutorial Set C - Part One

4. Find some example of typical bandwidth or link capacities by searching on the Internet.

- Capacity: Maximum number of bits that can be transmitted through a channel.
- Link: A connection between two nodes, which can be hosts, routers, or Ethernet switches.
- Link Capacity:
  - Nominal physical link – the maximum number of bits that the link can support. This is the upper bound of a link layer.
  - IP-layer link – The maximum number of IP-layer bits that can be transmitted from the source and correctly received by the destination over the link during the interval T to T+1, divided by I, where T is the time, and I a time interval.

Tutorial Set C - Part One

5. The classical DoS flood attack overwhelms the bandwidth of a link, to effectively shut that link down. Consider that we use ICMP pings of size 500 bytes. How many packets do we need to send per second to flood links with the following capacities?

a) 0.5 Mbps

b) 2 Mbps

c) 10 Mbps

Tutorial Set C - Part One

a) To flood a bandwidth of 0.5 Mbps with packets of length 500 bytes, the attacker would need at least (0.5 Mbps/(500 x 8) bits) of packets.

Thus, we need (0.5 x (1024 x 1024))/(500 * 8) <= 131.072 packets. In other words, we need **132 packets per second**.

Tutorial Set C - Part One

b) To flood a bandwidth of 2 Mbps with packets of length 500 bytes, the attacker would need at least (2 Mbps/(500 x 8) bits) of packets.

Thus, we need (2 x (1024 x 1024))/(500 * 8) <= 524.288 packets. In other words, we need **525 packets per second**.

Tutorial Set C - Part One

c) To flood a bandwidth of 10 Mbps with packets of length 500 bytes, the attacker would need at least (10 Mbps/(500 x 8) bits) of packets.

Thus, we need (10 x (1024 x 1024))/(500 * 8) <= 2621.44 packets. In other words, we need **2622 packets per second.**

Tutorial Set C - Part One

6. More problematic than the DoS attack is the distributed DoS attack. Assume each captured system has an upload capacity of 128-kbps. Assuming the same sized pings are used in the previous questions, how many such captured systems would be required to flood links with the following capacities?
   a) 0.5 Mbps
   b) 2 Mbps
   c) 10 Mbps

Tutorial Set C - Part One

a) From the previous question, we know that we need to send 132 packets to flood a link capacity of 0.5 Mbps.

If each computer has an upload capacity of 128 Kbps, we need at least (132 x 500 x 8)/(128 x 1024) computer, which is = 4.028 ≈ **5 computers**.

Tutorial Set C - Part One

b) From the previous question, we know that we need to send 525 packets to flood a link capacity of 2 Mbps.

If each computer has an upload capacity of 128 Kbps, we need at least (525 x 500 x 8)/(128 x 1024) computer, which is = 16.021 ≈ **17 computers**.

Tutorial Set C - Part One

c) From the previous question, we know that we need to send 2622 packets to flood a link capacity of 10 Mbps.

If each computer has an upload capacity of 128 Kbps, we need at least (2622 x 500 x 8)/(128 x 1024) computer, which is = 80.01 ≈ **81 computers**.

Tutorial Set C - Part One

7. What is a DNS?

DNS refers to Domain Name System. It is the **protocol** that defines the service that converts domain names to IP addresses. Its main objective is to be a mediator between the IP addresses, the system-side names of the websites and their respective domains, and their user-side alpha-numeric titles.

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2008, page 261 to 265.

Tutorial Set C - Part One

8. Describe DNS amplification.
   a) What implication does it have for the resources required by an attacker?
   b) How is DNS amplification different from general amplification, and how do they relate to a reflection attack?
   c) DNS amplification and reflection attacks do not generate backscatter traffic. Explain what backscatter traffic is and why this is the case.

Tutorial Set C - Part One

- DNS amplification is a type of DOS attack in which the attacker send a spoofed targeted address to a legitimate DNS server and use this server as the intermediary system to carry out the DOS attack.
- The attacker gain attack amplification by exploiting the behavior of the DNS protocol to convert a small request into a much larger response (amplification).

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2008, page 261 to 265.

Tutorial Set C - Part One

- In DNS amplification attack, the attacker creates a series of DNS requests containing the spoofed source address of the target system.
- These requests are directed at a number of the selected name servers.
- The servers respond to these requests, sending the replies to the spoofed source.
- The target is then flooded with the responses.

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2008, page 261 to 265.

Tutorial Set C - Part One

a) All that is needed by an attacker to carry out this attack is a name server with DNS records large enough for the amplification. (Note: with a classic DNS protocol, a 60-byte UDP request packet can result in a 512-byte UDP response. The more recent DNS protocol is able to extend to a much larger responses of over 4000 bytes, to support extended DNS features such as IPv6.)

   The attacker needs only generate a moderate flow of packets to cause a larger, amplified flow to flood and overflow the link to the target system.

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2008, page 261 to 265.

Tutorial Set C - Part One

b) With the general amplification, the attacker direct the original request to the broadcast address of some network. This resulted to all hosts on that network respond to the broadcasted request, and thus generating a flood of responses to the target. Thus the amplification is achieve through broadcasting.

With DNS amplification, the amplification is achieve through expanding the UDP responses.

Both general amplification and DNS amplification are a variant of reflector attacks that involve sending a packet with a spoofed source address for the target system to intermediaries.

William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2008, page 261 to 265.

Tutorial Set C - Part One

c) Backscatter is a side-effect of a spoofed DOS attack. In the course of a DOS attack, the attacker sends the victim large amount of network packets (IPpackets) that contain the spoofed address. The victim's machine is not able to differentiate the spoofed packets and the legitimate packets from its clients. So when the victim's machine responds, it sends its replies to the spoofed (falsified) address as well as its legitimate clients. These response packets (to the spoofed address) are known as backscatter.

Tutorial Set C - Part One

Backscatter does not happen to DNS amplification and reflection attacks because the attacker sends the network packets to normal functioning network servers. The servers respond to the packets, and sending their responses to the spoofed source address that belongs to the actual attack target. The fact that normal functioning server systems are being used as intermediaries, and that their handling of the packets is entirely conventional, the responses from the victim are also normal.

Tutorial Set C - Part One

9. NuCaptcha: What is it? You should go to the website of the company and have a look at some examples. Look at the article in the lab directory too and see what claims they make.

   It is a video-based CAPTCHA that can adapt to easy challenges for legitimate users and difficult ones for attackers. NuCaptcha uses measure indicators to identify high risk activity and combats it with a real-time adaptive Captcha.

http://en.wikipedia.org/wiki/NuCaptcha

Tutorial Set C - Part One

10. DeCaptcha: What is it?

http://www.ubergizmo.com/2011/05/decaptcha-defeats-captchas/

Tutorial Set C – Part Three

1. What is the expected cost of calculating a puzzle consisting of m k–bit puzzles, in the Client Puzzle Protocol of Juels and Brainard (1999)?

The expected cost of calculating a puzzle is $m \times 2^{k-1}$.

Tutorial Set C – Part Three

2. Using the above formula, draw up a table demonstrating the cost for $1 \leq m \leq 20$, $1 \leq k \leq 60$.

TutorialSetC-Part3-Q2-PuzzleCost-2015S43

Tutorial Set C – Part Three

3. Assume a uniform distribution of hash values for the hash function used.
   a) What is the probability of a single guess by an attacker solving the puzzle, as a function of m and k?
   b) Using the above formula, draw up a table demonstrating the probability for $1 \leq m \leq 20$, $1 \leq k \leq 60$.
   c) What if the distribution of hash values wasn't uniform? Would it increase or decrease the probabilities determined above?

Tutorial Set C – Part Three

a) What is the probability of a single guess by an attacker solving the puzzle, as a function of m and k?

The probability for an attacker to solve the puzzle in a single guess is $\left(1/2\right)^{km}$.

Tutorial Set C – Part Three

b) Using the above formula, draw up a table demonstrating the probability for $1 \leq m \leq 20$, $1 \leq k \leq 60$.

TutorialSetC-Part3-Q3b-Probability-2015S43

Tutorial Set C – Part Three

c) What if the distribution of hash values wasn't uniform? Would it increase or decrease the probabilities determined above?

If the distribution of hash values was not uniform, then there is likely to have duplicate hash values which may increase the probability or the likely hood of getting the puzzle guessed correctly.

Tutorial Set C – Part Three

4. Why use multiple puzzles rather than one single large one?

If we use a single big puzzle instead of sub puzzles, then the difficulty level is hard to adjust. This is because one bit of change in k could require a much longer time to solve the puzzle. Using sub puzzles we can fine tune the difficulty level.

Tutorial Set C – Part Four

3. Construct a puzzle with two sub-puzzles, that involves no hashing in the generation of the puzzle.

Tutorial Set C – Part Four

Server:

Selects Ns, Ns'

Sends (1, Ns) and (2, Ns') to Client

Client:

Finds x such that hash(1, C, Ns, x) = 111…111Y, where 111…111 denotes k 1's, and Y can be any value.

Finds x' such that hash(2, C, Ns', x') = 111…111Y', where 111…111 denotes k 1's, and Y can be any value.

Sends (1, C, Ns, x) and (2, C, Ns', x') to Server

Server checks if there are k 1's in each solution.

Spring 2016 – Part B, Question 3

Question 1 …1

1) One of the client puzzles we considered contained the statement $h(C, Ns, Nc, Y) = 000 \ldots 000X$.
   a. Describe each of the components in the expression above.
   b. How much work is required to "solve" the puzzle, in the context of this statement?
   c. What is the purpose of such a puzzle?
   d. Describe how we could modify this to generate sub-puzzles.
   e. What advantage do we obtain by using many sub-puzzles rather than just one single large puzzle?

Question 1 …2

$$h(C, Ns, Nc, Y) = 000 \ldots 000X.$$

a) Describe each of the components in the expression above.

Question 1 …2

$$h(C, Ns, Nc, Y) = 000 \ldots 000X.$$

a) Describe each of the components in the expression above.

- h: a cryptographic hash function (e.g., MD5 or SHA)
- C: the client identity
- N_s: the server's nonce
- N_c: the client's nonce
- Y: the solution of the puzzle
- $000 \ldots 000$: the k first bits of the hash value; must be zero. The reasonable values of k lie between 0 and 64.
- X: the rest of the hash value; may be anything

Question 1 …3

b) How much work is required to “solve” the puzzle, in the context of this statement?

Question 1 …3

b) How much work is required to "solve" the puzzle, in the context of this statement?

The cost of solving the puzzle depends exponentially on the required number of k of zero bits in the beginning of the hash. If $k = 0$, no work is required. If $k = 64$, then in the worst case, it would be 2^k. In such a puzzle, the reasonable values of k lie between 0 and 64.

Question 1 …4

c) What is the purpose of such a puzzle?

Question 1 …4

c) What is the purpose of such a puzzle?

The purpose of such a puzzle is to ensure that the client should always commit its resources to the authentication protocol first and the server should be able to verify the client commitment before allocating its own resources.

Question 1 …5

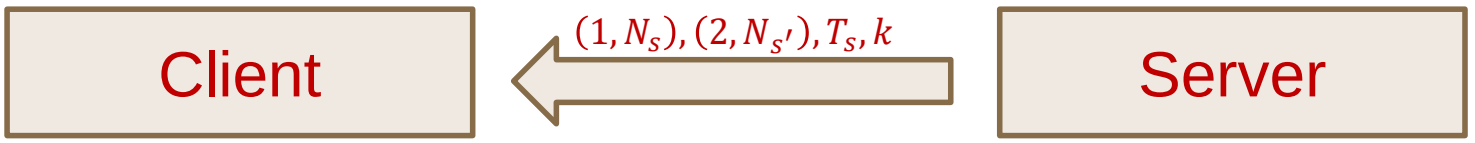
d) Describe how we could modify this to generate sub-puzzles.

Question 1 …5

d) Describe how we could modify this to generate sub-puzzles.

The modification can be done as follow:
The Server:

- Server determines k.
- Server generates two nonce - $N_s \; and \; N_{s'}$, and a timestamp T_s.
- Server sends the two puzzles with sequence number - $(1, N_s) \; and \; (2, N_{s'})$, the timestamp T_s and the puzzle difficulty level k to client.



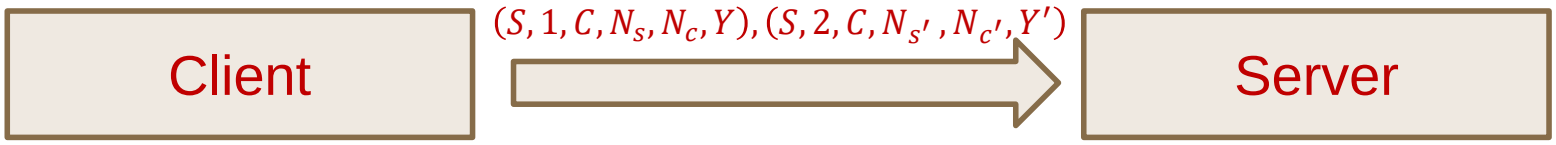
Question 1 …6

The Client:

- Client receive the puzzles and commits its resources into solving the puzzle.
- Client verifies the timestamp T_s.
- Client generates two nonce N_c *and* $N_{c'}$
- Client finds Y such that $h(1, C, N_s, N_c, Y) = 000 \ldots 000X$, where $000 \ldots 000$ denotes k $0's$, and X can be any value.
- Client finds the second solution Y' such that $h(2, C, N_{s'}, N_{c'}, Y') = 000 \ldots 000X$, where $000 \ldots 000$ denotes k $0's$, and X can be any value.

Question 1 …7

- Client sends the solutions (solved puzzles) $(S, 1, C, N_s, N_c, Y)$ and $(S, 2, C, N_s', N_c', Y')$ to server.



Question 1 …8

The Server:

- verifies that $N_s \; and \; N_{s'}$ are recent.
- checks that $C, N_s, N_{s'}, N_c, and \; N_{c'}$ have not been used before.
- checks if there are $k \; 0's$ in each solution, that is, the server checks if $h(1, C, N_s, N_c, Y) = 000 \ldots 000X$ and $h(2, C, N_{s'}, N_{c'}, Y') = 000 \ldots 000X$ are correct.
- If they do, the server commits the resources, stores $(C, N_s, N_{s'}, N_c, N_{c'})$ and sends $(S, C, N_c, N_{c'})$ to the client.
- The operation can now continue.

Question 1 …9

e) What advantage do we obtain by using many sub-puzzles rather than just one single large puzzle?

Question 1 …9

e) What advantage do we obtain by using many sub-puzzles rather than just one single large puzzle?

If we use a single big puzzle instead of sub puzzles, then the difficulty level is hard to adjust. This is because one bit of change in *k* could require a much longer time to solve the puzzle. Using sub puzzles we can fine tune the difficulty level.

Reference

- William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson Education, 2012

- 3a_SandhuET1996

- http://en.wikipedia.org/wiki/NuCaptcha