Security and Real World HTTP Servers

LIGHTHOUSE LABS

AGENDA

- Security
- REST
- Revisiting Middleware

LIGHTHOUSE LABS

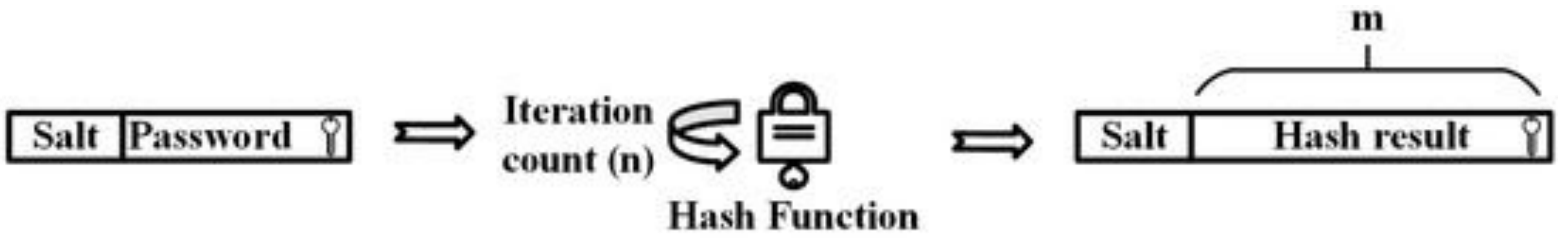
Security

Security issue #1: passwords

- Passwords that are stored in plain text represent a security breach
- What's the solution?

Hashing

Hashing



- Hashing is a one way process
- Each iteration is doubling the time it takes to hash the content
- We simply need to increase the iteration count if computers become more powerful

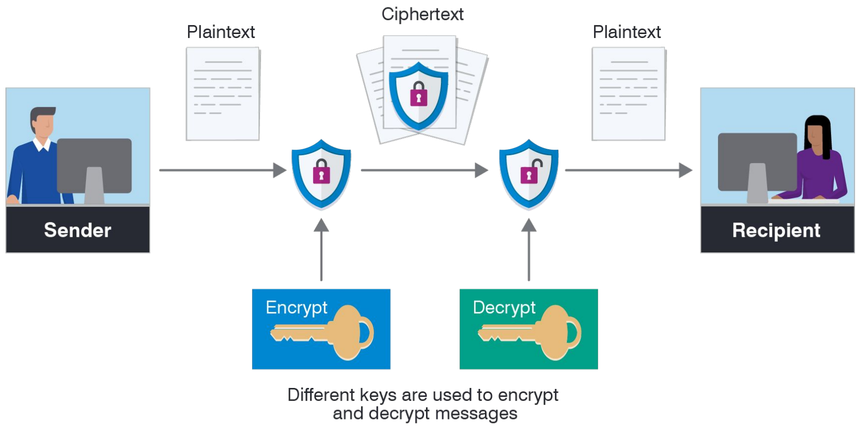
Security

Security issue #2: cookies

- Cookie information is stored in plain-text
- Plain text cookies can be modified
- We might impersonate another user
- What’s the solution?

Encryption

Encryption

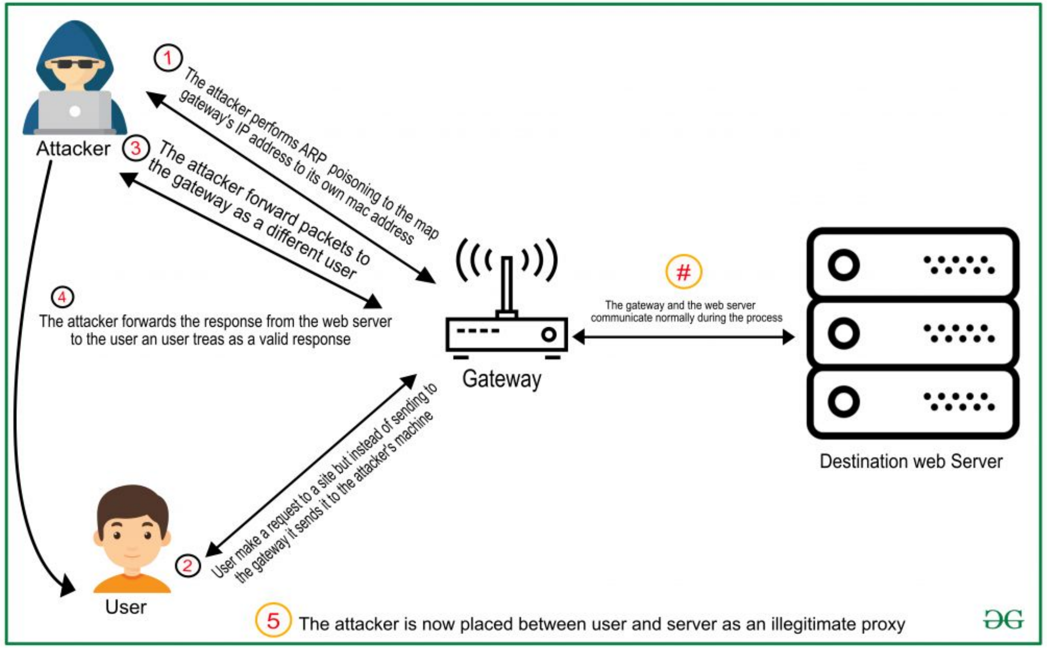


- Information is encrypted using a key
- It can be decrypted by the recipient using the key

Image credit: https://www.proofpoint.com/

Security

Security issue #3: stealing cookies



Attacker
1 The attacker performs ARP poisoning to the map gateway's IP address to its own mac address
3 The attacker forward packets to the gateway as a different user
4 The attacker forwards the response from the web server to the user an user treas as a valid response
Gateway
# The gateway and the web server communicate normally during the process
Destination web Server
2 User make a request to a site but instead of sending to the gateway it sends it to the attacker's machine
User
5 The attacker is now placed between user and server as an illegitimate proxy

- HTTP is plain-text
- Man-in-the-middle attack
- What's the solution?

https

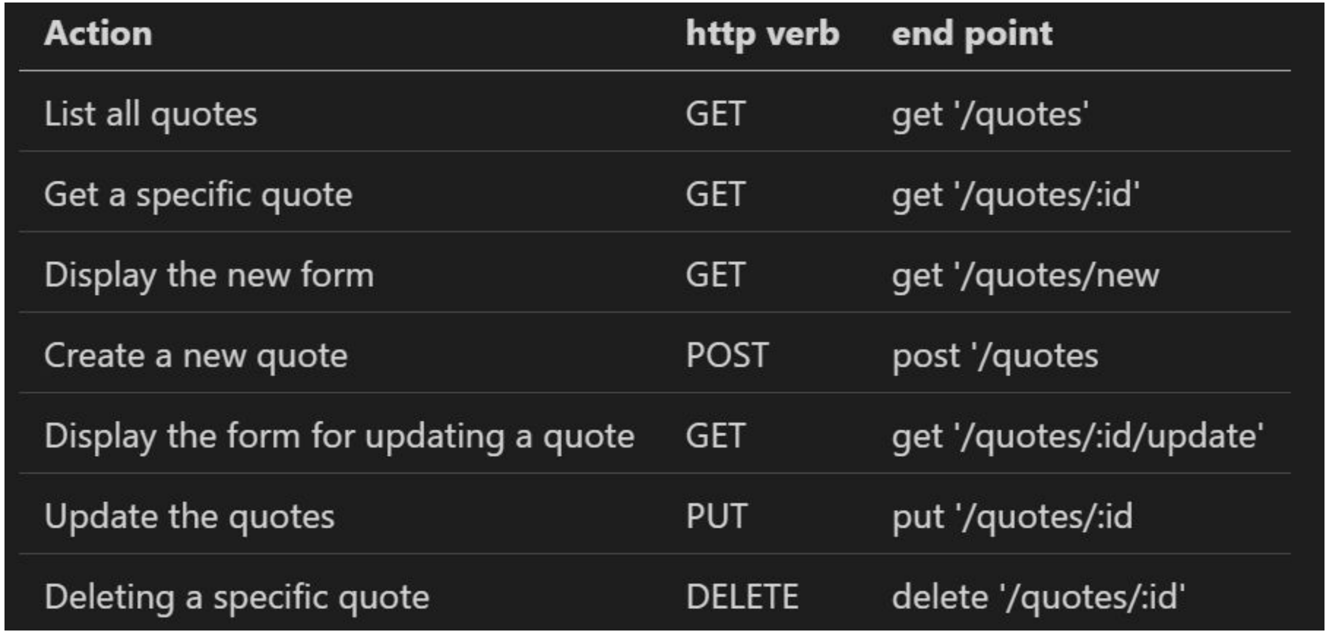
REST

Representational State Transfer

- REST is a pattern, a convention to organize our url structure
- Resource based routes convention (The key abstraction of information in REST is a resource
- It should use http verbs to express what the request wants to accomplish
- Resource information must be part of the url

REST

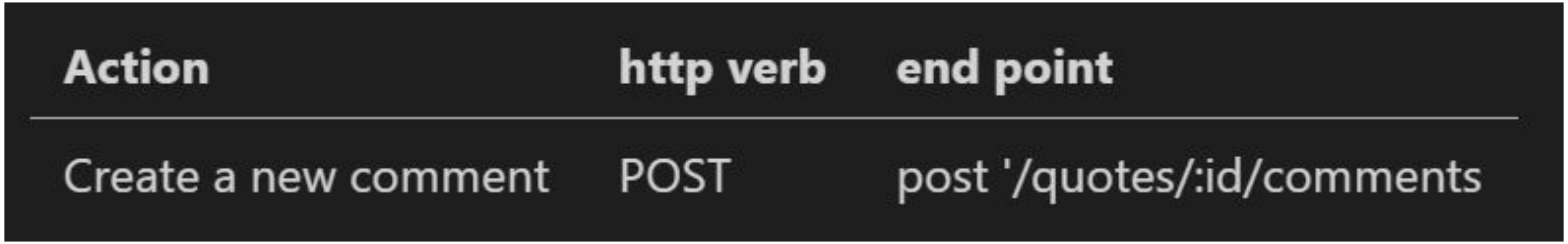
By following REST principles, it allows us to design our end points:



Action	http verb	end point
List all quotes	GET	get '/quotes'
Get a specific quote	GET	get '/quotes/:id'
Display the new form	GET	get '/quotes/new
Create a new quote	POST	post '/quotes
Display the form for updating a quote	GET	get '/quotes/:id/update'
Update the quotes	PUT	put '/quotes/:id
Deleting a specific quote	DELETE	delete '/quotes/:id'

REST

Nested Resources:

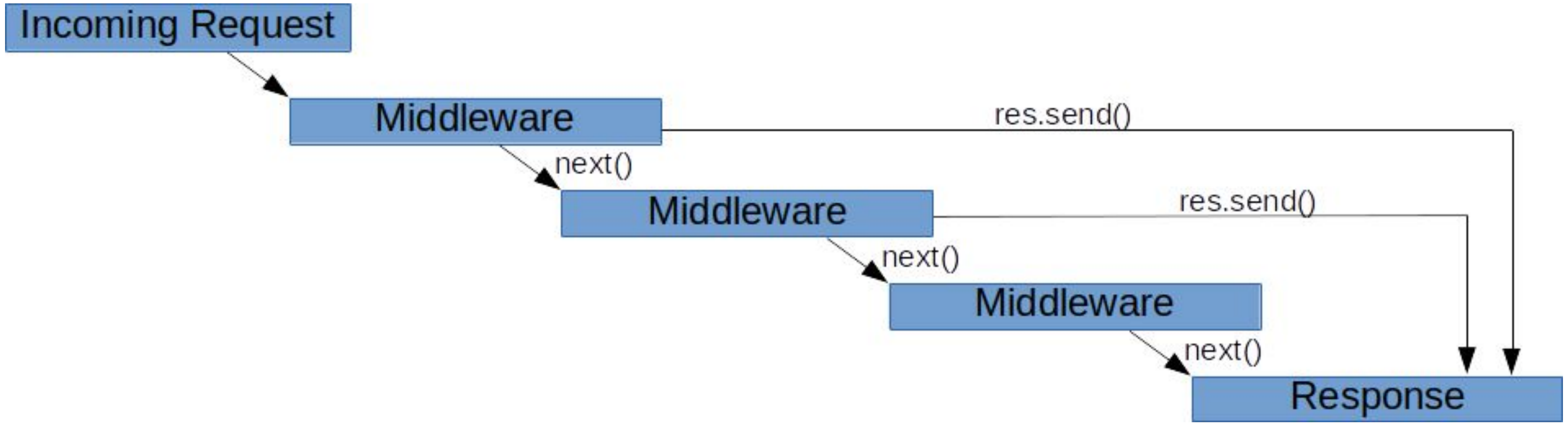


Action	http verb	end point
Create a new comment	POST	post '/quotes/:id/comments

REST Exercise: https://gist.github.com/DominicTremblay/941afbe1295ec666d3539d448df7c776

Revisiting Middleware

Middleware is a piece of software that sits in **between** the request and the response.



Questions?

LIGHTHOUSE LABS