Secure Content Delivery Using Amazon CloudFront

Eric Chu, eCloudvalley Digital Technology

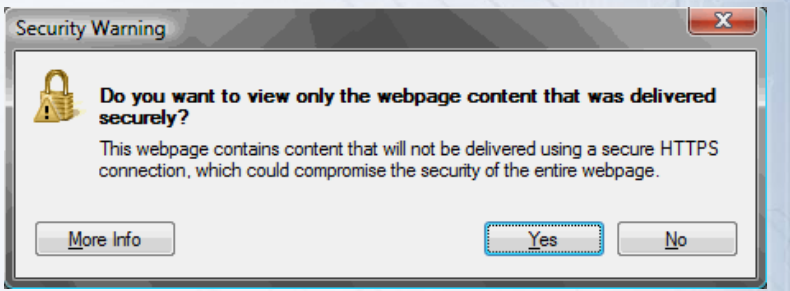
What to Expect from the Session

In this session we will talk about:

- Why security matters
- Key aspects of security
- How Amazon CloudFront can help
- Best practices for secured delivery on Amazon CloudFront

Overview: Why Security Matters

- Customer Trust



- Regulatory Compliance

- Data Privacy

eCloud valley

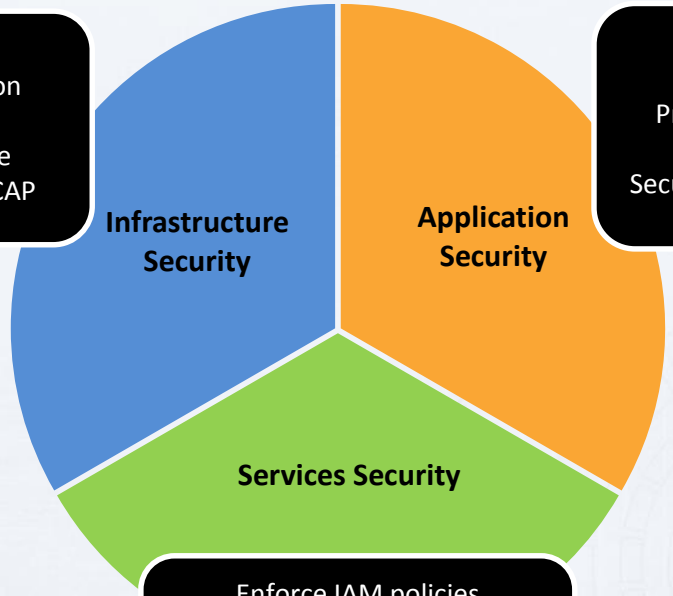
How AWS Can Help

In the cloud, security is a shared responsibility

https://aws.amazon.com/compliance/shared-responsibility-model/

SOC 1,2,3
ISO 27001/2 Certification
PCI DSS 2.0 Level 1-5
HIPAA/SOX Compliance
FedRAMP, FISMA & DIACAP
ITAR

How we secure our infrastructure



Encrypt data in transit
Encrypt data at rest
Protect your AWS Credentials
Rotate your keys
Secure your application, OS, Stack and AMIs

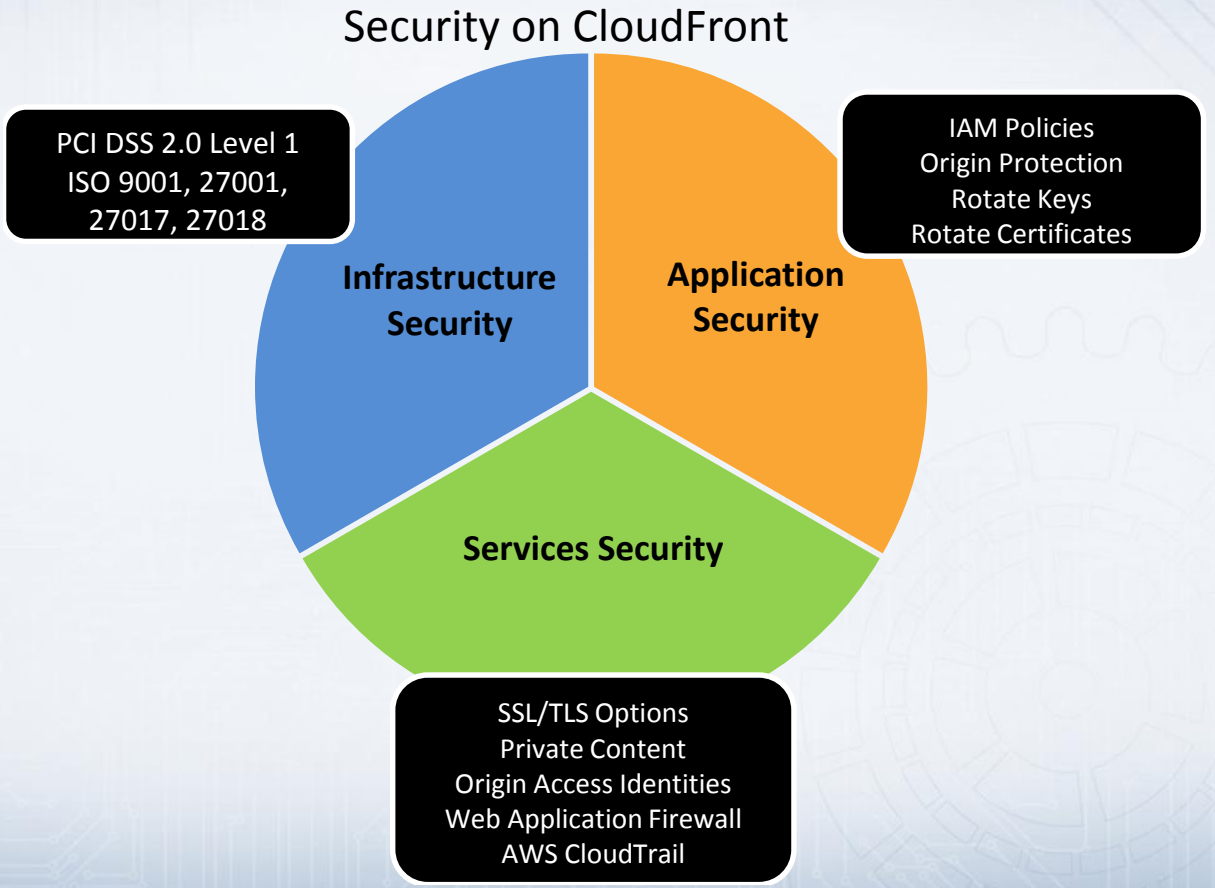
How can you secure your application?

What security options and features are available to you?

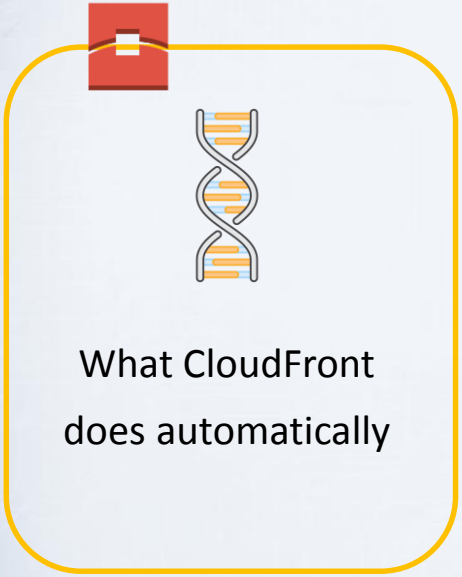
Enforce IAM policies
Use MFA, VPC, Leverage S3 bucket policies
EC2 Security groups
EFS in EC2, ACM, etc.

eCloud valley

How CloudFront Can Help



How CloudFront Can Help



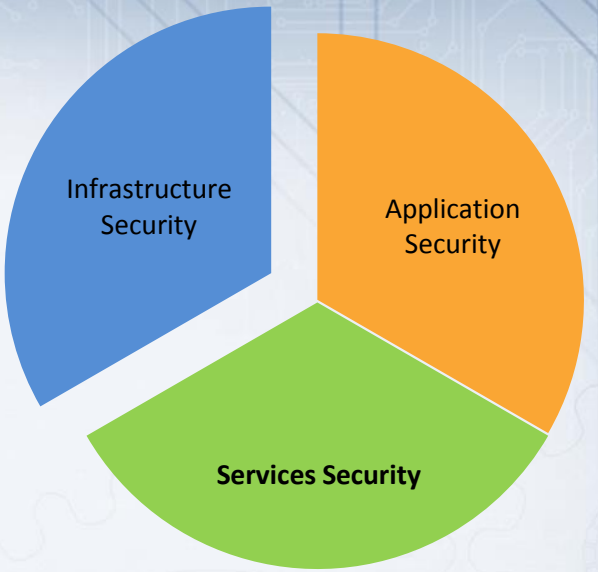
What CloudFront does automatically

+

What you can do using CloudFront features

=

Highly secure content delivery

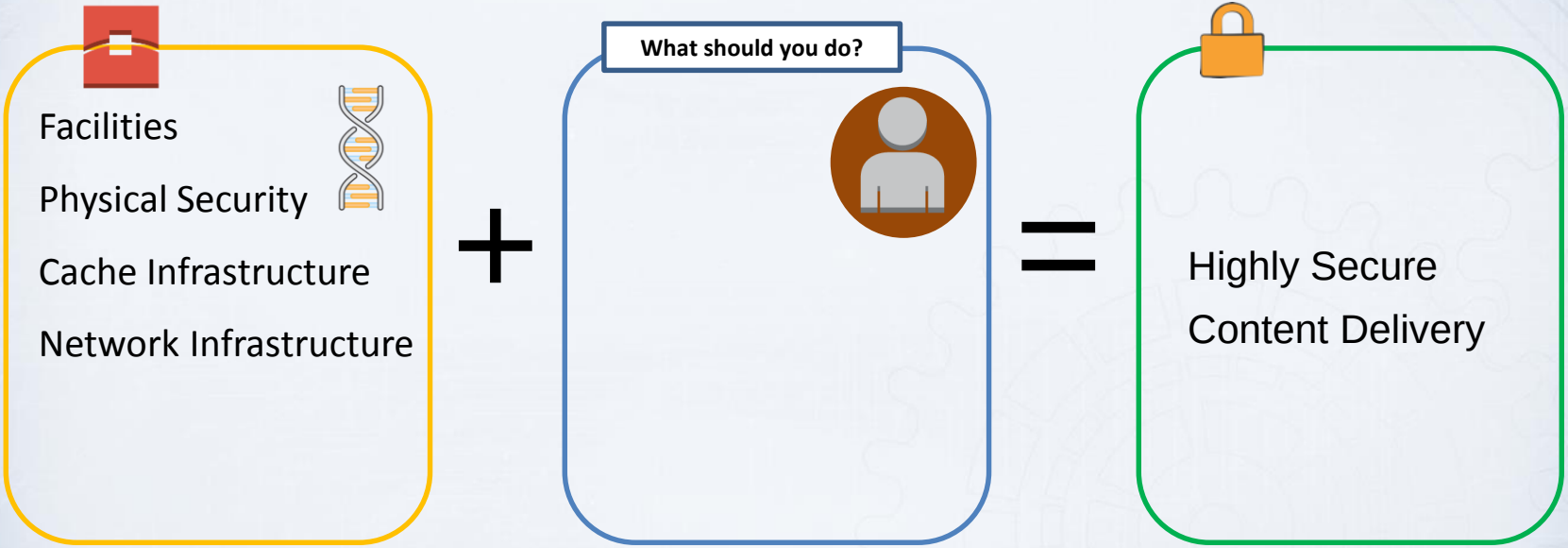


Infrastructure Security

How we secure our infrastructure

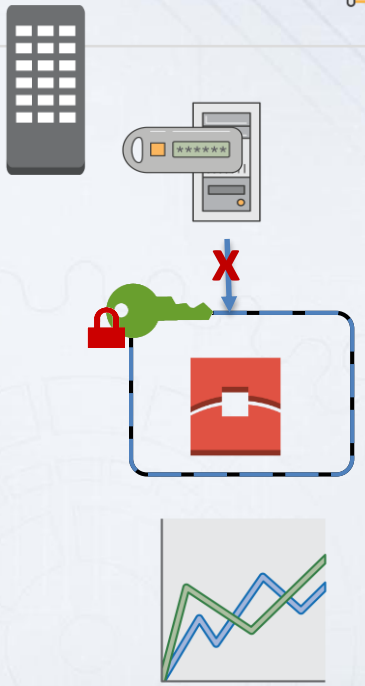
eCloud valley

Infrastructure Security



Infrastructure Security

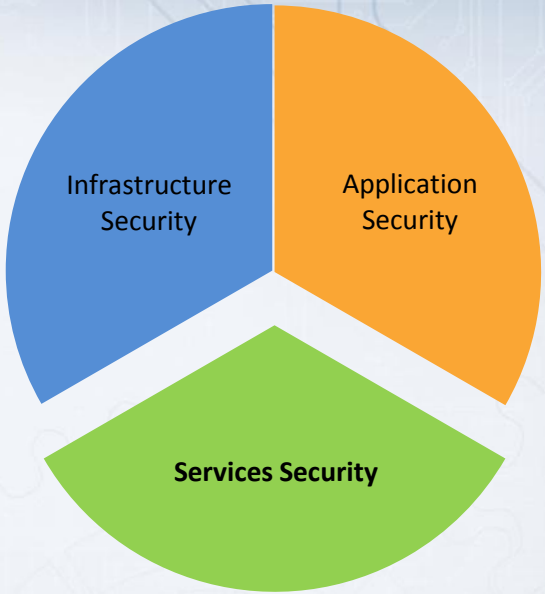
- Bastion hosts for maintenance
- Two-factor authentication
- Encryption
- Separation to enhance containment
- Testing & metrics



eCloud valley

Infrastructure Security

Services Security



Security options and features available on CloudFront

eCloud valley

Services Security

High Security Ciphers

PFS

OCSP Stapling

Session Tickets

+

What should you do?

SSL/TLS Options

Private Content

Trusted Signers

Web Application Firewall

AWS CloudTrail

AWS Certificate Manager

=

Highly Secure Content Delivery

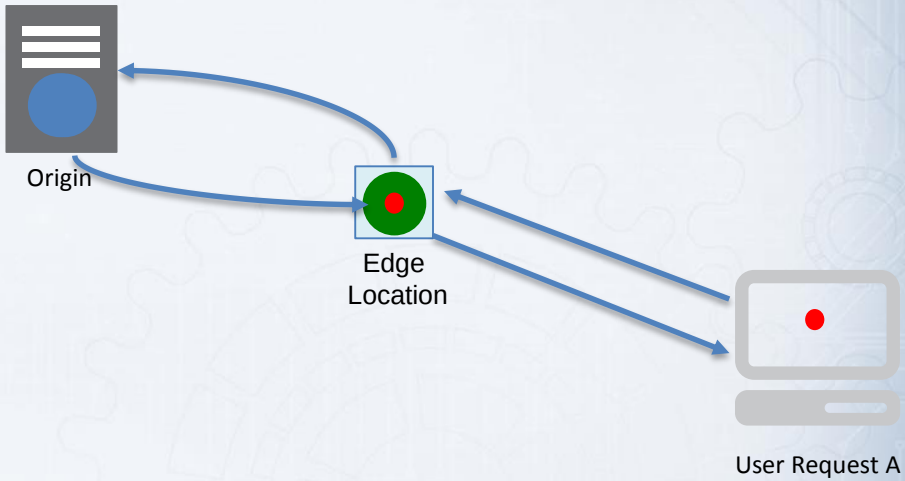
eCloud valley

CloudFront can protect ‘Data in Transit’

eCloud valley

CloudFront Protects Data in Transit

- Deliver content over HTTPS to protect data in transit
- HTTPS Authenticates CloudFront to Viewers
- HTTPS Authenticates Origin to CloudFront



eCloud valley

CloudFront enables advanced SSL features automatically

Advanced SSL/TLS

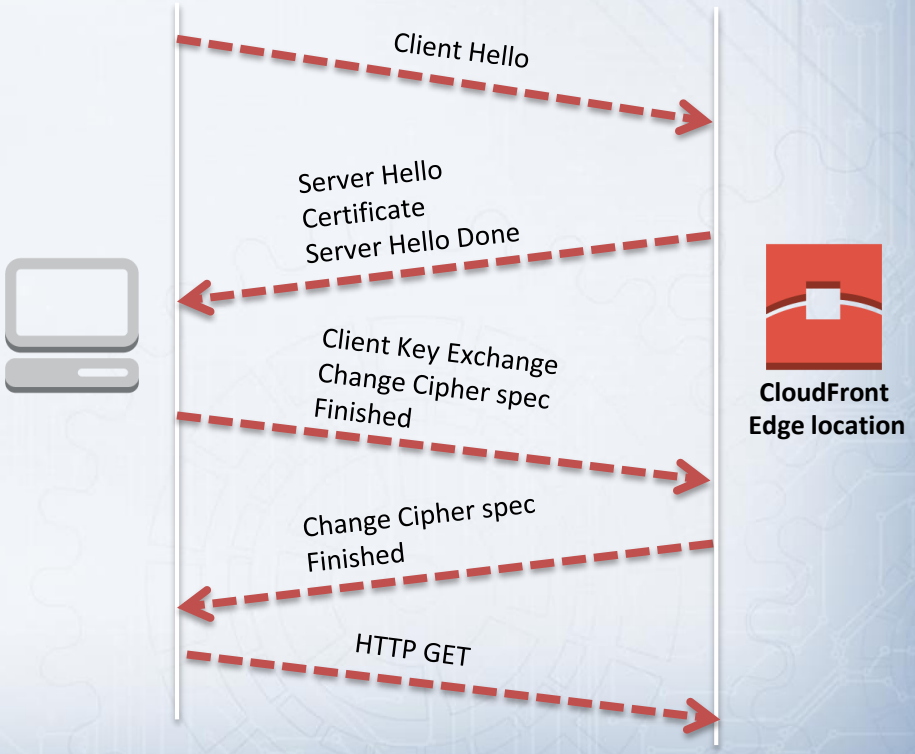
- **Improved Security**
- High security ciphers
- Perfect Forward Secrecy

- **Improved SSL Performance**
- Online Certificate Status Protocol (OCSP Stapling)
- Session Tickets

eCloud valley

Advanced SSL/TLS: Improved Security

- CloudFront uses high security ciphers
- Employs ephemeral key exchange
- Enables Perfect Forward Secrecy



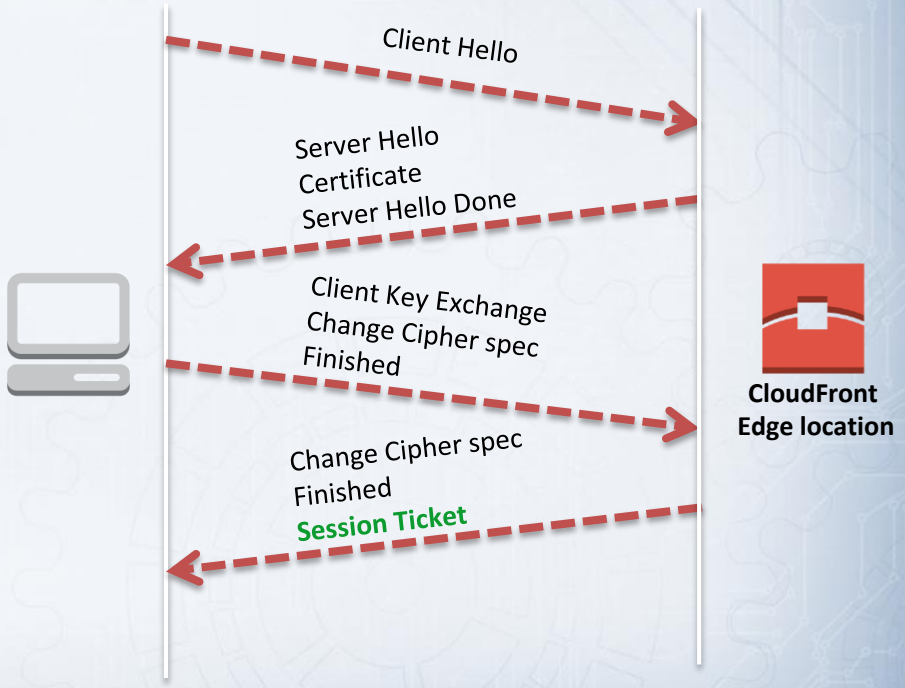
Advanced SSL/TLS: Improved Performance

- Session Tickets

- Online Certificate Status Protocol (OCSP Stapling)

Session Tickets

- Session tickets allow client to resume session
- CloudFront sends encrypted session data to client
- Client does an abbreviated SSL handshake



OCSP Stapling

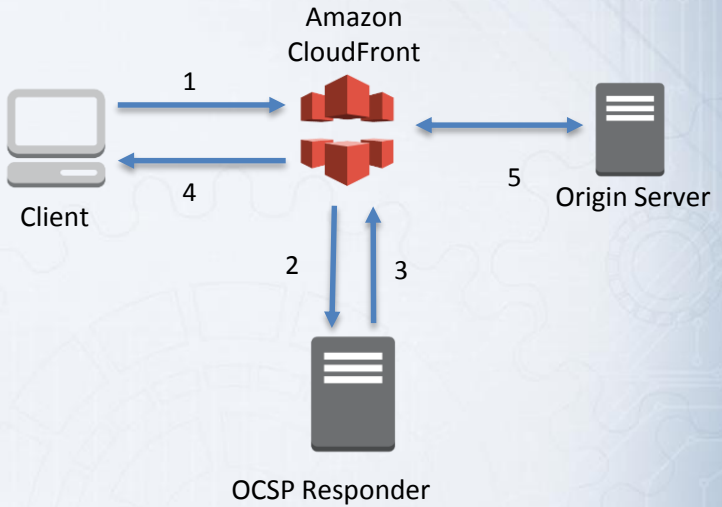
1) Client sends TLS Client Hello

2) CloudFront requests certificate status from OCSP responder

3) OCSP responder sends certificate status

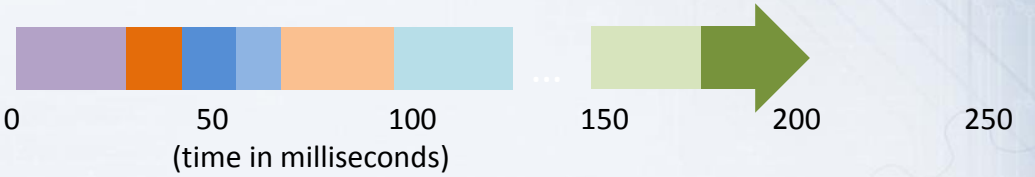
4) CloudFront completes TLS handshake with client

5) Request/response from Origin Server

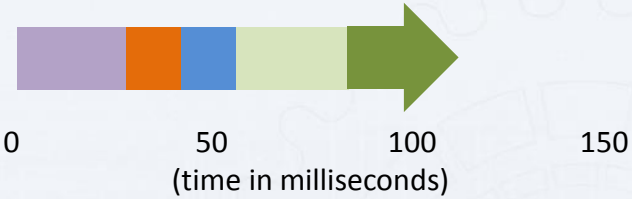


OCSP Stapling

Client Side Revocation Checks



OCSP Stapling



120 ms faster

30% Improvement

TCP Handshake	DNS for OCSP Responder	… Follow Certificate Chain
Client Hello	TCP to OCSP Responder	Complete Handshake
Server Hello	OCSP Request/Response	Application Data

eCloud valley

Validate Origin Certificate

- CloudFront validates SSL certificates to origin

✓ Origin domain name must match Subject Name on certificate

✓ Certificate must be issued by a Trusted CA

✓ Certificate must be within expiration window

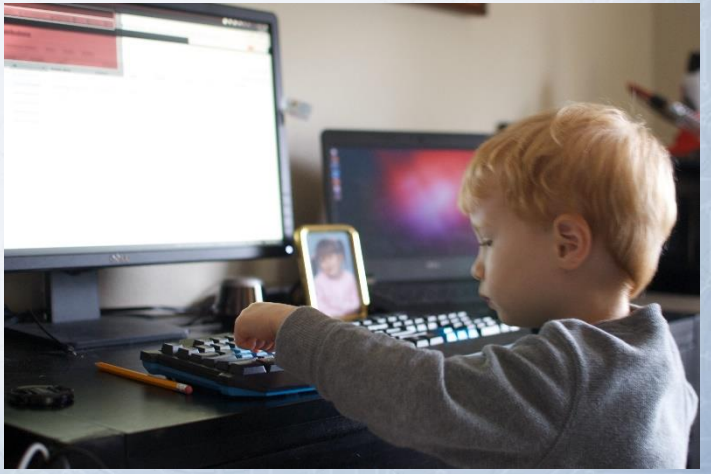
eCloud valley

But there are things you need to do

eCloud valley

Deliver Content using HTTPS

- CloudFront makes it easy
- Create one distribution, and deliver both HTTP & HTTPS content
- There are other options as well:
  - Strict HTTPS
  - HTTP to HTTPS redirect



CloudFront TLS Options

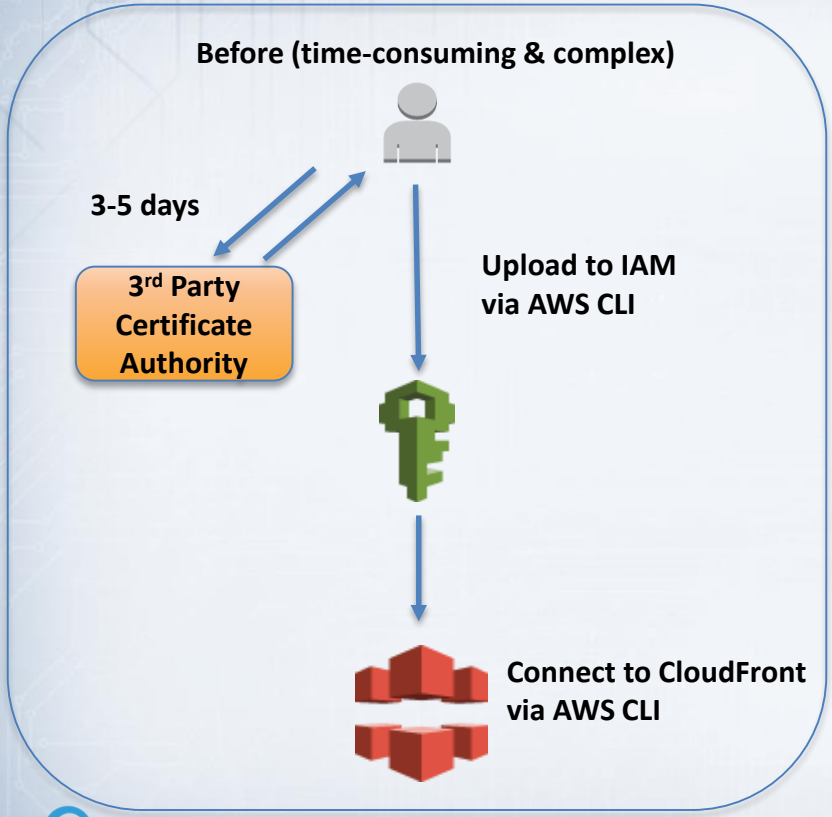
- Default CloudFront SSL Domain Name
- CloudFront certificate shared across customers
- *When to use?*
- *Example: dxxx.cloudfront.net*

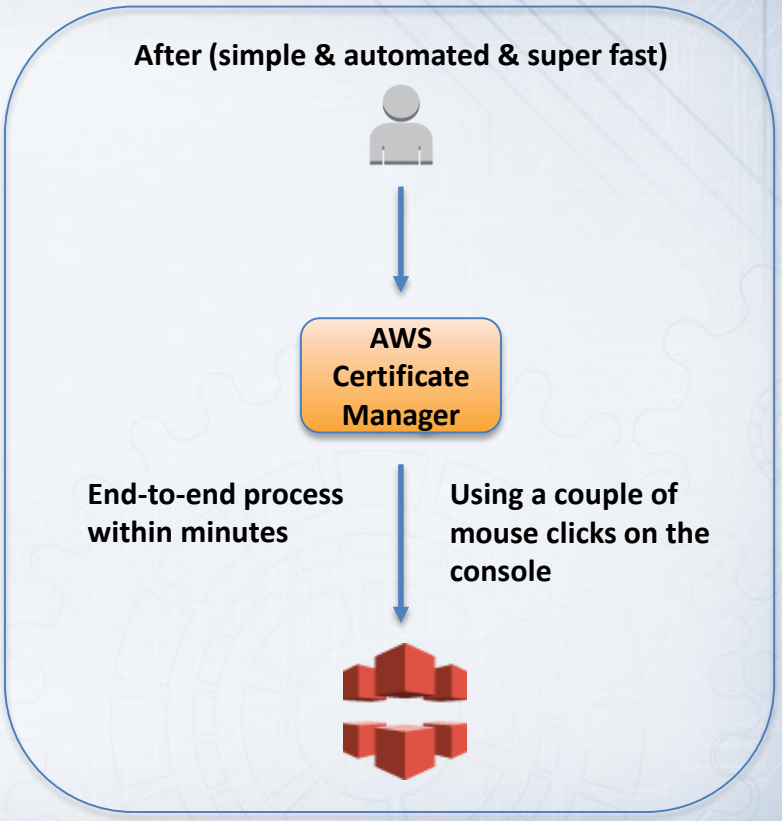
- SNI Custom SSL
- Bring your own SSL certificate
- OR use AWS Certificate Manager
- Relies on the SNI extension of the Transport Layer Security protocol
- *When to use?*
- *Example: www.mysite.com*
- Some older browsers/OS do not support SNI extension

- Dedicated IP Custom SSL
- Bring your own SSL certificate
- OR use AWS Certificate Manager
- CloudFront allocates dedicated IP addresses to serve your SSL content
- *When to use?*
- *Example: www.mysite.com*
- Supported by all browsers/OS

eCloud valley

Integrated with AWS Certificate Manager





eCloud valley

MapBox

MapBox uses SNI Custom SSL

- They wanted to use a custom domain
  - xxxxx.mapbox.com
- Their clients support TLS
- They wanted to use an economical option

eCloud valley

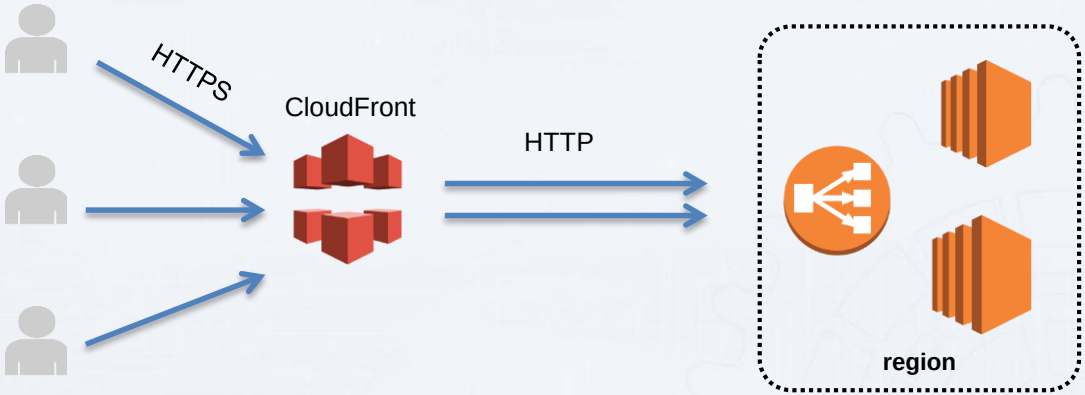
HTTPS Usage Patterns

- Half bridge TLS termination
- Full bridge TLS termination

eCloud valley

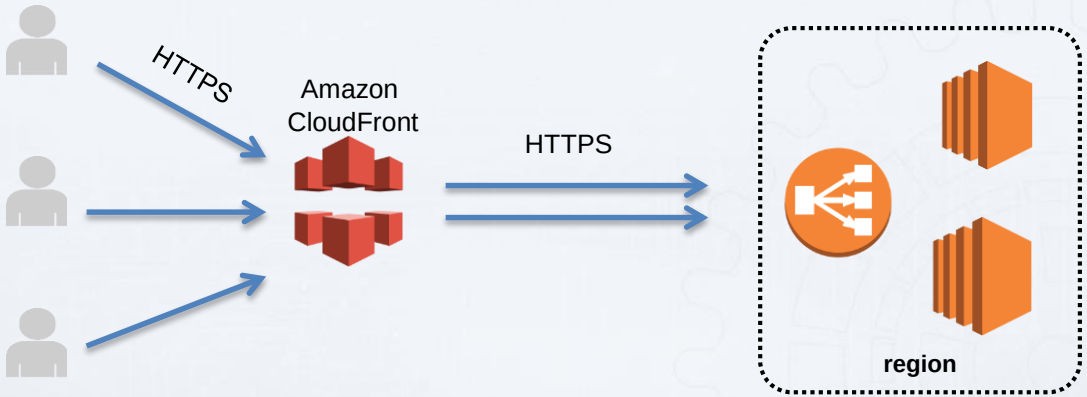
Half Bridge TLS Termination

- Better performance by leveraging HTTP connections to origin



Full Bridge TLS Termination

- Secured connection all the way to origin
- Use origin ‘Match Viewer’ or ‘HTTPS Only’



MapBox uses multiple origins

- Have multiple API endpoints (origin servers)
- One with Half Bridge: HTTP from Edge to Origin
- Second with Full Bridge: HTTPS from Edge to Origin

eCloud valley

You are not done yet…

You need to protect content cached at the Edge

eCloud valley

Access Control

- What if you want to…

- Deliver content only to selected customers
- Allow access to content only until ‘time n’
- Allow only certain IPs to access content

Access Control: Private Content

• Signed URLs

- Add signature to the Querystring in URL
- Your URL changes

• When should you use it?

- Restrict access to individual files
- Users are using a client that doesn't support cookies
- You want to use an RTMP distribution

• Signed Cookies

- Add signature to a cookie
- Your URL does not change

• When should you use it?

- Restrict access to multiple files
- You don’t want to change URLs

eCloud valley

Access Control: Private Content

- Here is an example of a policy statement for signed URLs

```
{
   "Statement": [{
      "Resource":"URL or stream name of the object",
      "Condition":{
         "DateLessThan":{"AWS:EpochTime":required ending date and time in Unix time format and UTC},
         "DateGreaterThan":{"AWS:EpochTime":optional beginning date and time in Unix time format and UTC},
         "IpAddress":{"AWS:SourceIp":"optional IP address"}
      }
   }]
}
```

Access Control: Private Content

- Under development mode?

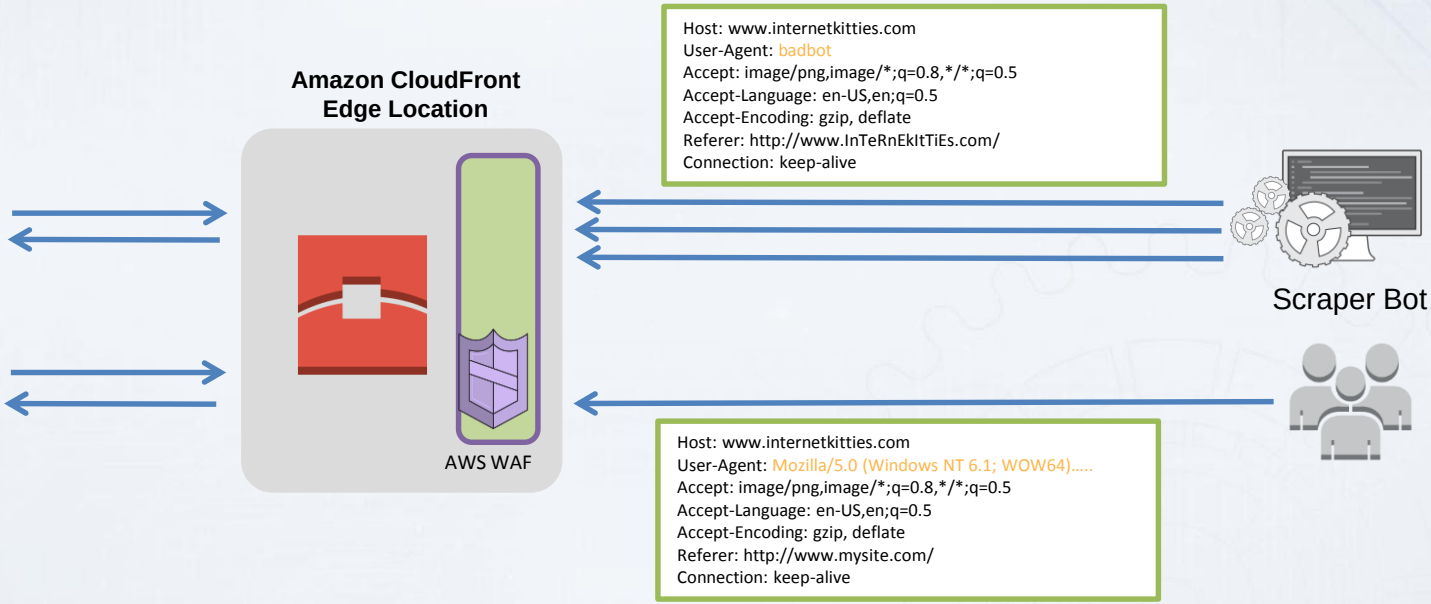
- Make CloudFront accessible only from your “Internal IP Addresses”

You are still not done…

What if you want to restrict access based on parameters in the request?

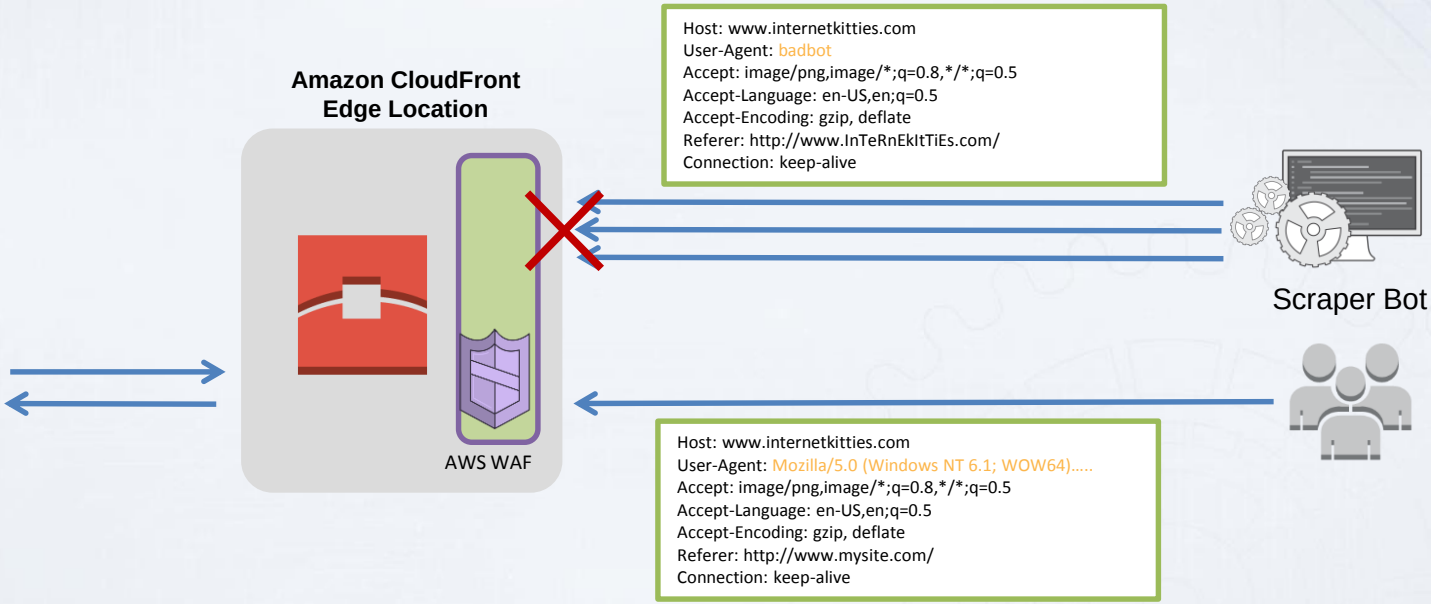
eCloud valley

Serving Unnecessary Requests Costs Money

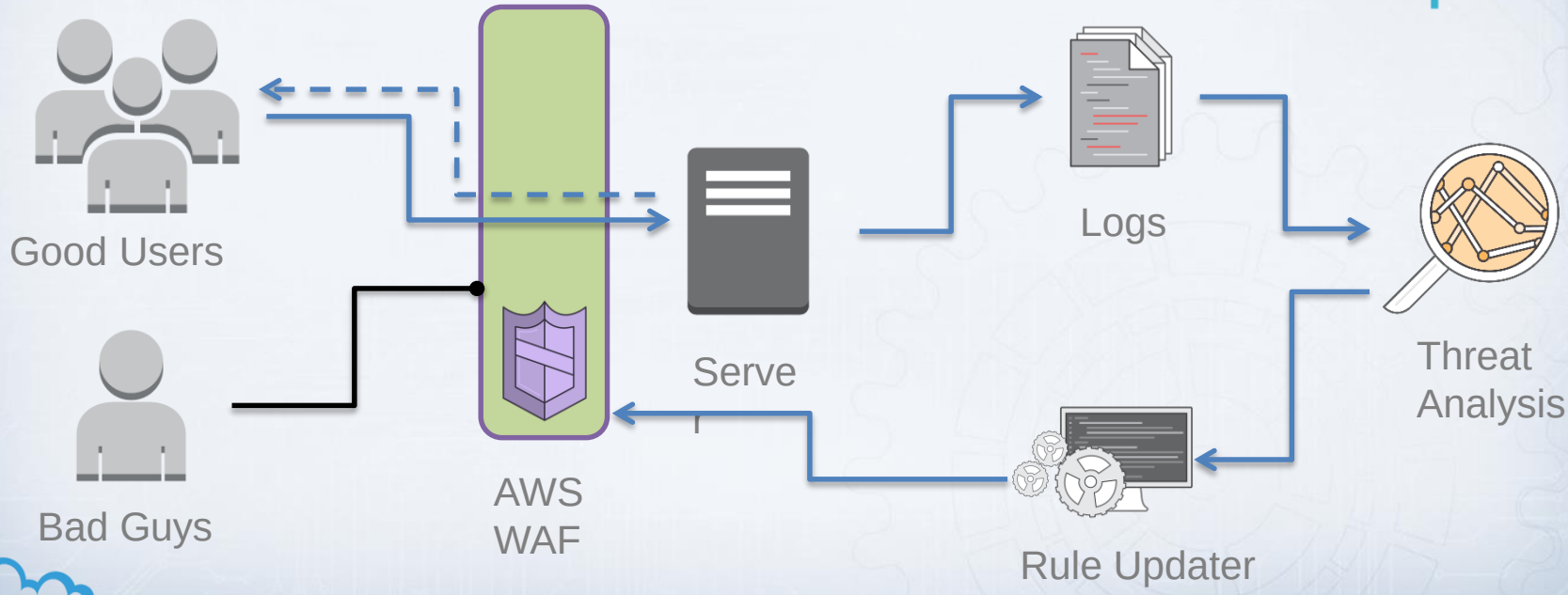


eCloud valley

Access Control: Web Application Firewall



MapBox uses AWS WAF to protect from bots



eCloud valley

AWS WAF example: A technical implementation

- Blocking bad bots dynamically with AWS WAF web ACLs

AWS WAF Example: Blocking Bad Bots

- What We Need…
- **IPSet**: contains our list of blocked IP addresses
- **Rule**: blocks requests if requests match IP in our **IPSet**
- **WebACL**: allow requests by default, contains our **Rule**

- and…
- Mechanism to **detect** bad bots
- Mechanism to **add** bad bot **IP address** to **IPSet**

eCloud valley

AWS WAF Example: Detecting Bad Bots

- Use **robots.txt** to specify which areas of your site or webapp should not be scraped
- Place file in your web root
- Ensure there are links pointing to non-scrapable content
- Hide a trigger script that normal users don't see and good bots ignore

```
$ cat webroot/robots.txt
User-agent: *
Disallow: /honeypot/
```

```
<a href="/honeypot/" class="hidden" aria-hidden="true">click me</a>
```

eCloud valley

AWS WAF Example: Blacklist Bad Bots

- Bad bots (ignoring your robots.txt) will request the hidden link
- Trigger script will detect the source IP of the request
- Trigger script requests change token
- Trigger script adds source IP to **IPSet** blacklist
- **WebACL** will block subsequent request from that source

```
$ aws --endpoint-url https://waf.amazonaws.com/ waf get-change-token
{
  "ChangeToken": "acbc53f2-46db-4fbd-b8d5-dfb8c466927f"
}
$ aws --endpoint-url https://waf.amazonaws.com/ waf update-ip-set --cli-input-json '{ "IPSetId": "<<IP SET ID>>", "ChangeToken": "acbc53f2-46db-4fbd-b8d5-dfb8c466927f", "Updates": [ { "Action": "INSERT", "IPSetDescriptor": { "Type": "IPV4", "Value": "<<SOURCE IP>>/32" } } ] }'
{
  "ChangeToken": "acbc53f2-46db-4fbd-b8d5-dfb8c466927f"
}
```

eCloud valley

Preconfigured Protection & Tutorials

https://aws.amazon.com/waf/preconfiguredrules/

AWS WAF Tutorials

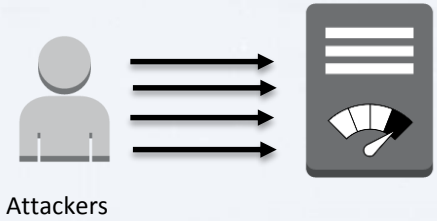
Blocking IP Address that Exceed Request Limits: one security challenge you may have faced is how to prevent your web servers from being affected by distributed denial of service (DDoS) attack, commonly called HTTP floods. In this tutorial, you will provision a solution that will identify IP addresses that are sending requests over your defined threshold and updates your AWS WAF rules to automatically block subsequent requests from those IP addresses.

Get Started with Blocking IP Addresses that Exceed Request Limits

Blocking IP Addresses that Submit Bad Requests: Internet-facing web applications are frequently scanned by various sources, and unless managed by you, the sources probably don't have good intentions. To find vulnerabilites, these scans send out a series of requests that generate HTTP 4xx error codes which you can use to identify and block. In this tutorial, you'll create a Lambda function that automatically parses CloudFront access

Types of attacks that need automation

IP reputation lists

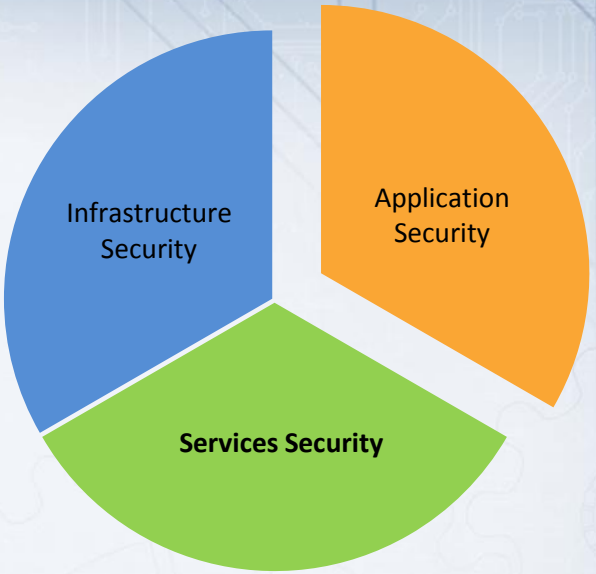


HTTP floods

Scans & probes

Bots & scrapers

Application Security

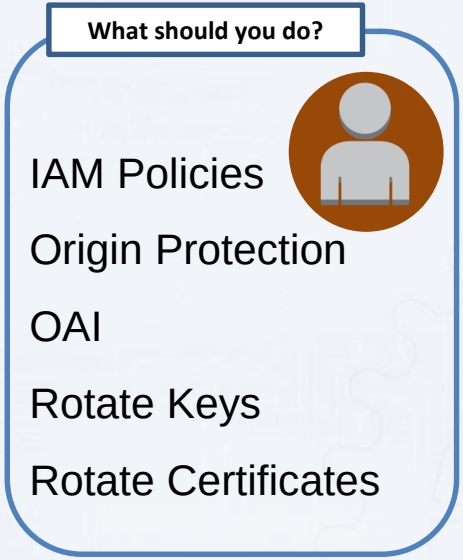


How can you secure your application and origin?

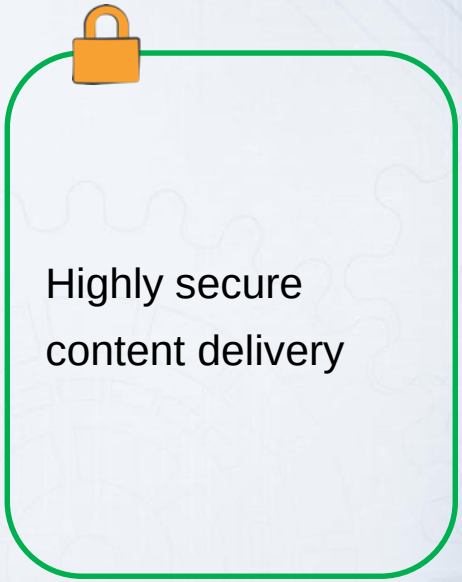
eCloud valley

Application Security

+



=



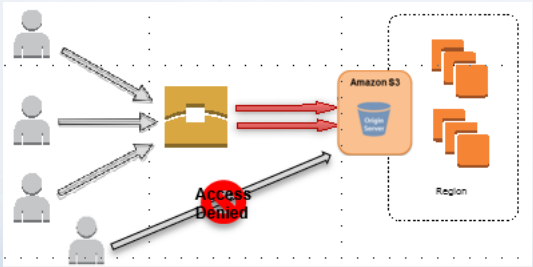
eCloud valley

Hackers could still bypass CloudFront to access your origin…

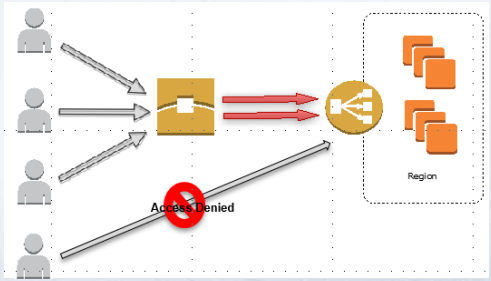
eCloud valley

Access Control: Restricting Origin Access

- Amazon S3
  - Origin Access Identify (OAI)
    - Prevents direct access to your Amazon S3 bucket
    - Ensures performance benefits to all customers



- Custom Origin
  - Block by IP Address
  - Pre-shared Secret Header
    - Whitelist only CloudFront
    - Protects origin from overload
    - Ensures performance benefits to all customers

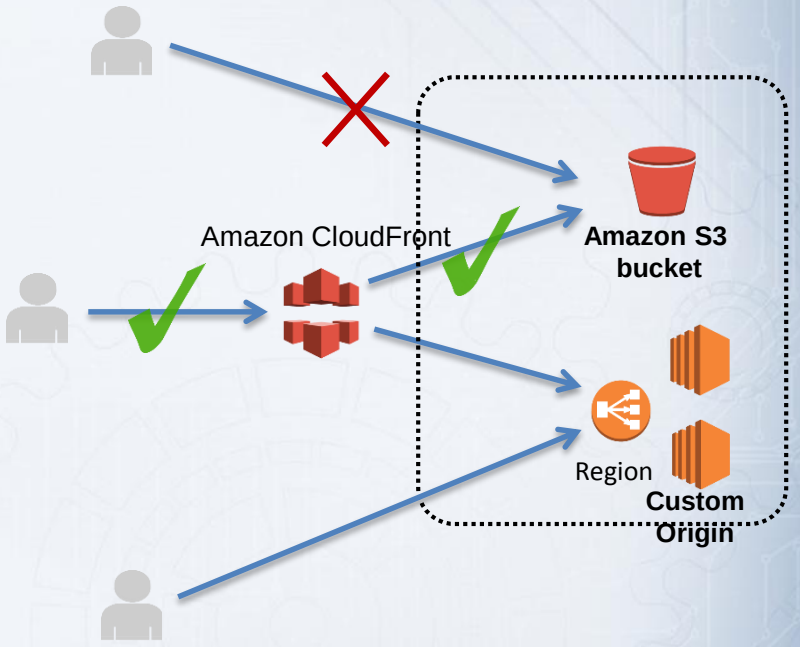


eCloud valley

Object Access Identity (OAI)

- Only CloudFront can access Amazon S3 bucket

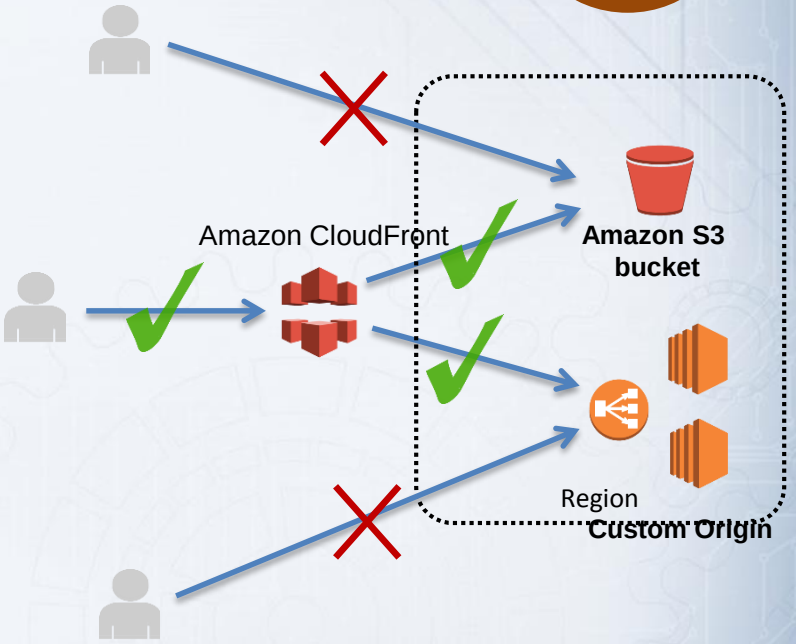
- We make it simple for you



Shield Custom Origin

1. Whitelisting CloudFront IP Range

2. Whitelist a pre-shared secret origin header



Shield Custom Origin

- Subscribe to SNS notifications on changes to IP ranges
- Automatically update security groups
- https://github.com/awslabs/aws-cloudfront-samples

eCloud valley

Services Security: IAM

- AWS Managed Policies or create custom policies
- Regulate access to CloudFront APIs
- Describe user role or permissions

eCloud valley

Origin Best Practices

- 1. Match Viewer Origin Protocol Policy
- Enable Only TLS 1.1 or 1.2 to Origin
- Enforce HTTPS Only Connections to Origin

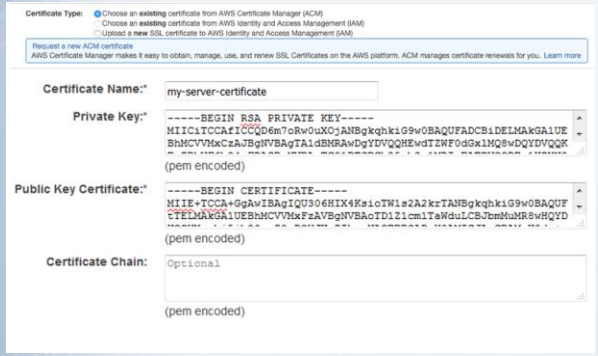
- 2. Restrict Access using Security Groups & Shared Secret

- 3. Use a SHA256 certificate

eCloud valley

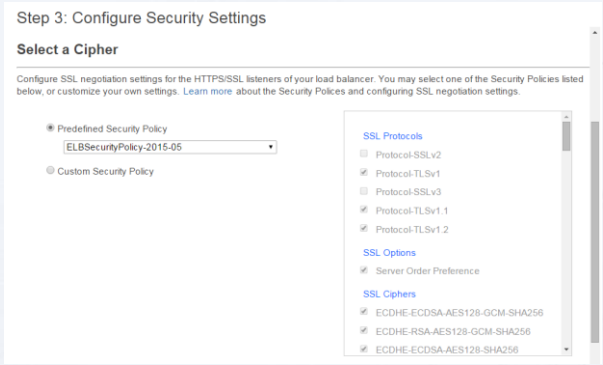
Origin Best Practices

- 4. Use ELB with Custom certificate



You can request an SSL Certificate from AWS Certificate Manager

- 5. Use ELB Pre-defined Policy



- 6. Send HSTS Header

- `*Strict-Transport-Security: max-age=15552000;`
- `*X-Frame-Options: SAMEORIGIN`
- `*X-XSS-Protection: 1; mode=block Options`

eCloud valley

Services Security : IAM Examples

- Example 1: Create groups with just access to create invalidations

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Stmt1442882368000",
            "Effect": "Allow",
            "Action": [
                "cloudfront:CreateInvalidation"
            ],
            "Resource": [
                "*"
            ]}]}
```

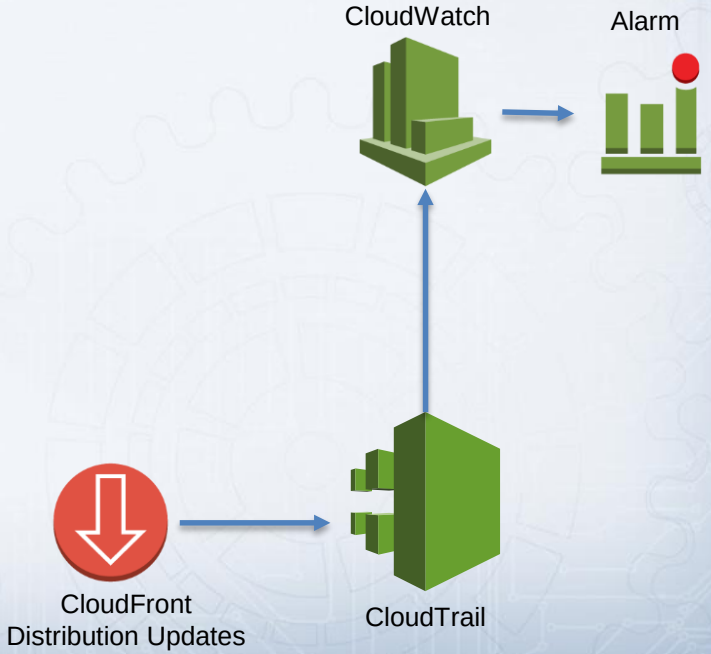
- Example 2: Just read access to your distributions & configuration

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Stmt1442882731000",
            "Effect": "Allow",
            "Action": [
                "cloudfront:Get*",
                "cloudfront:List*"
            ],
            "Resource": [
                "*"
            ]}]}
```

eCloud valley

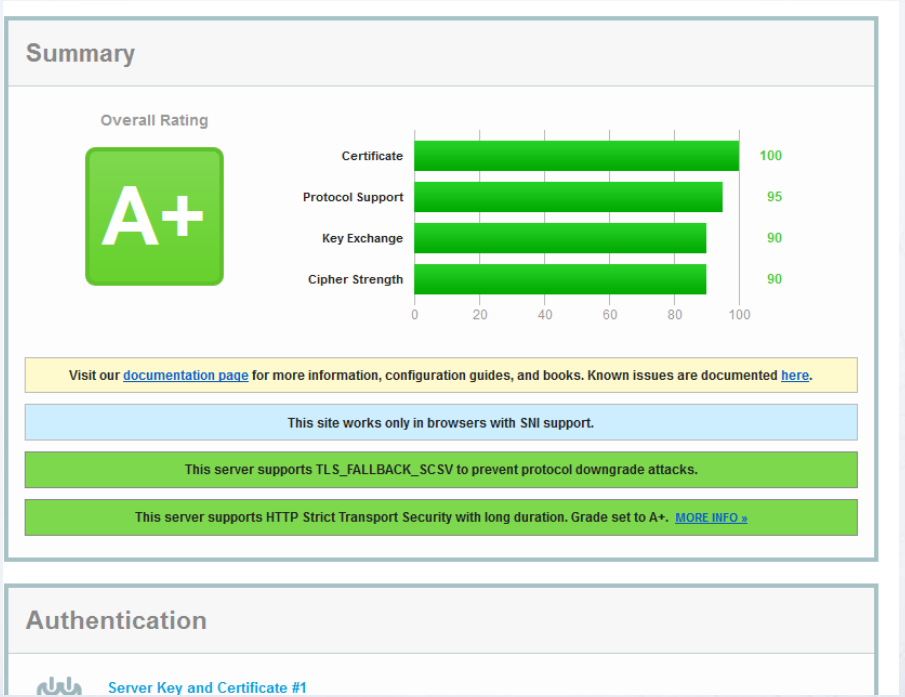
AWS CloudTrail

- Record CloudFront API calls history for:
- Security analysis
- Resource change tracking
- Compliance auditing



eCloud valley

How to validate your security configurations



https://www.ssllabs.com/ssltest/

Thank You