AWS re:Inforce

JUNE 10 – 12, 2024 | PHILADELPHIA, PA

IAM321

Amazon S3 presigned URL security

Bryant Cutler

Principal Engineer
AWS

20
8
2
19

Amazon S3 presigned URLs

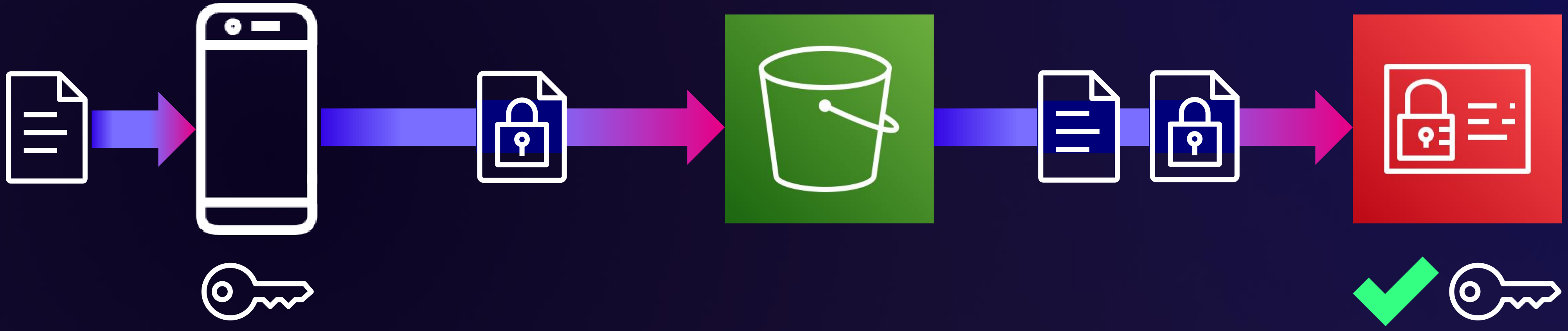
How they work

When to use them

Alternatives

AWS authentication basics

Is the request signature correct?



Is the key used to create the request signature the same as the key identified in the request?

Is the request signature valid?

Is the access key used in the signature currently active?

Is the request signer from an AWS account in good standing?

Is the timestamp used in the signature close to the current server time?

So what is a presigned URL?

presigned URLs

≠

signed requests

Amazon S3 presigned URLs

Relax the timestamp matching constraint

Include an expiration time parameter

Allows for a range of valid time of use

Signed parameters cannot be updated by the client

Maximize convenience for URL users

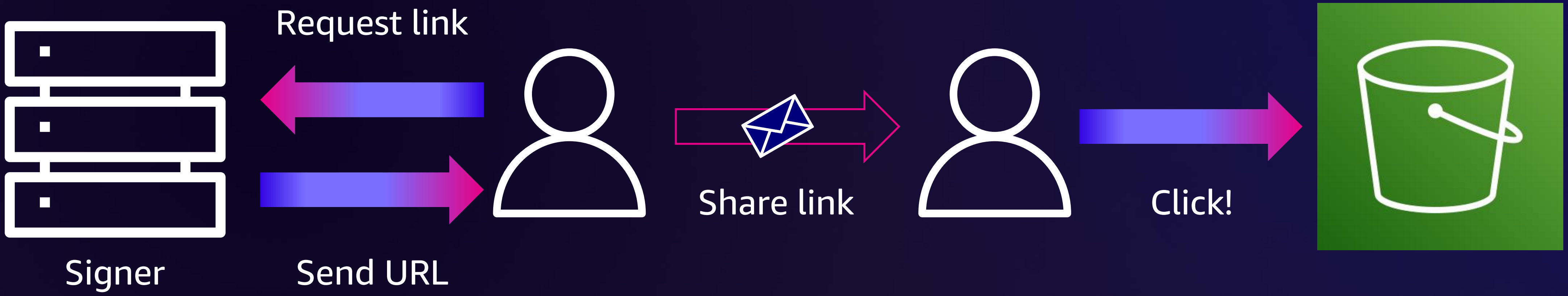
Send request parameters as URL query parameters

For uploads, do not sign request body, just the object key

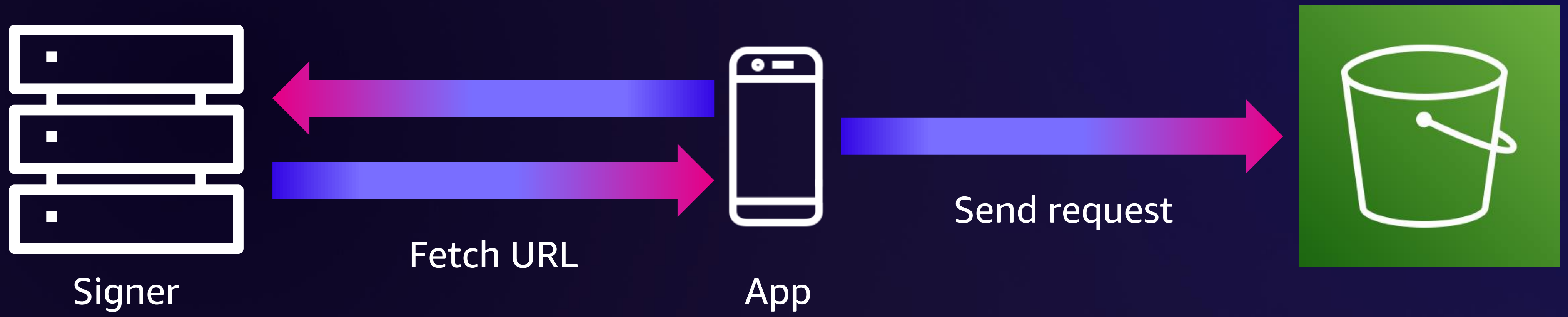
Example

https://reinforce-demo.s3.us-east-1.amazonaws.com/access_grants.svg?response-content-disposition=attachment&X-Amz-Security-Token=XXXX&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20240515T210625Z&X-Amz-SignedHeaders=host&**X-Amz-Expires=300**&X-Amz-Credential=ASIA3FKOA5G4ACL7YSF2%2F20240515%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Signature=d4560284ba8b9a801f403290add28ac4999639fecef7dd0338cbe9faee8c8b56

Presigned URLs for humans



Applications can use presigned URLs too



When to use presigned URLs

Client constraints

Client constraints

Generic user agent

Client constraints

Generic user agent

No AWS identity

Client constraints

Generic user agent

No AWS identity

Per-object control

Why not use presigned URLs?

Duration is limited by credential lifetime

Authenticates a single **specific** API call on a specific object

Upload URLs don’t allow for object checksum values

Presigned URLs are **bearer tokens**

Bearer tokens?

Anyone can use them, subject to the signer permissions

No audit record of the user, only the signer

Replayable within their duration

Revokable only by impacting signer credentials/permissions

So, should I be afraid of presigned URLs?

NO

Mitigate presigned URL risks

1. Carefully bound signer permissions
2. Don’t log request signatures
3. Keep bearer token durations short
4. Use temporary sessions for signing

Mitigate presigned URL risks

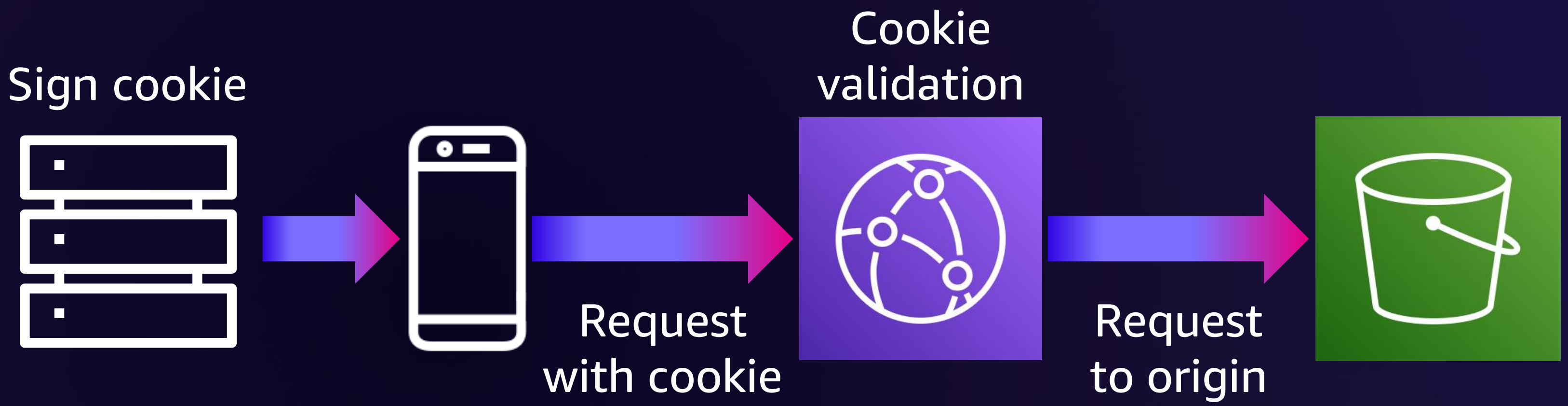
1. Carefully bound signer permissions
2. Don’t log request signatures
3. Keep bearer token durations short
4. Use temporary sessions for signing

Security-minded customers will be following these best practices already!

Alternatives

Amazon CloudFront signed cookies

A GREAT ALTERNATIVE FOR END USER-FACING LINKS



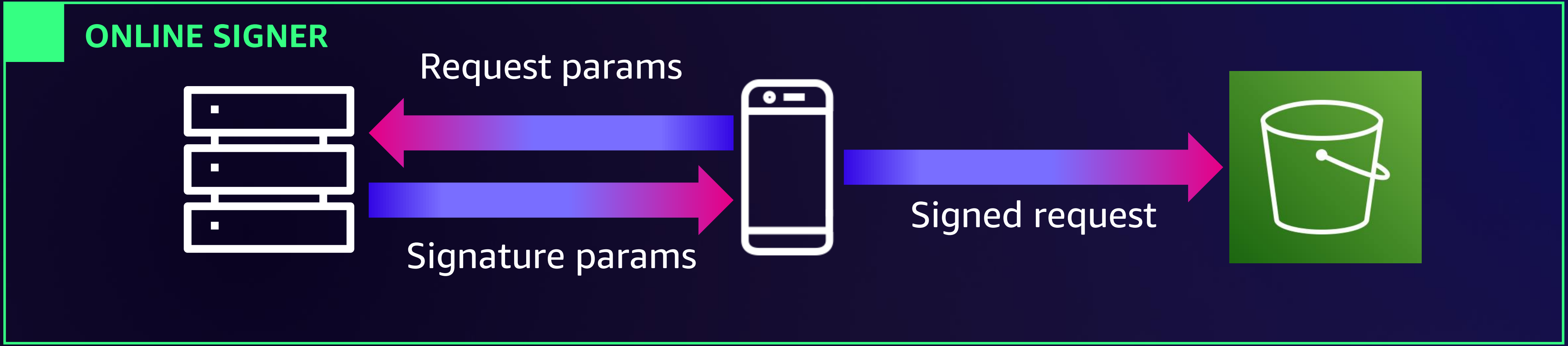
Custom domain name support with TLS

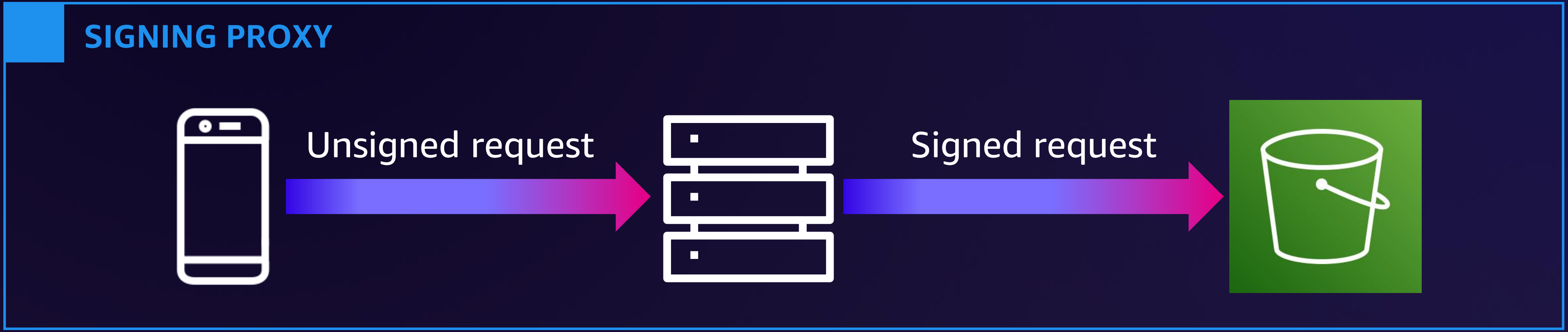
Can allow access to multiple files

Secured by browsers from cross-site access in local cookie storage

Request signers

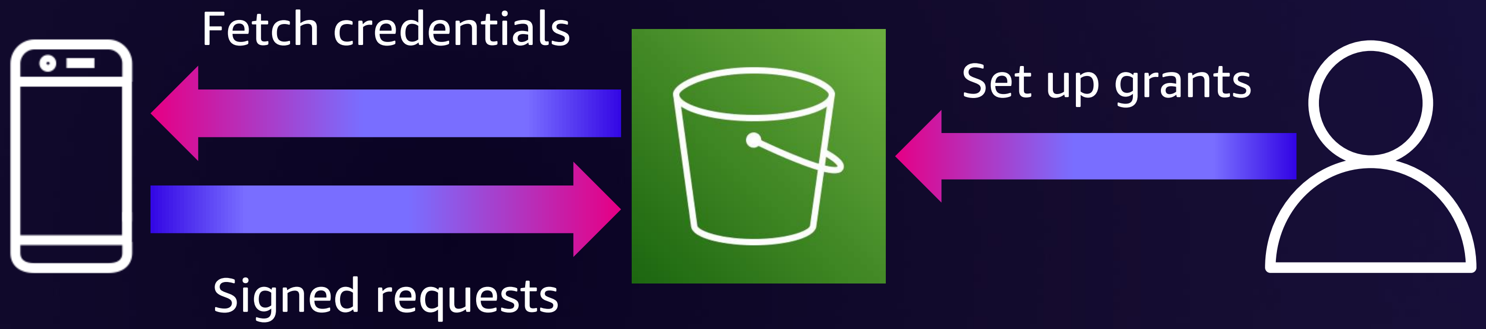
ENABLE SHORT-LIVED ACCESS FOR APPLICATIONS


ONLINE SIGNER


SIGNING PROXY

Amazon S3 Access Grants

FLEXIBLE AND SCALABLE CREDENTIAL VENDOR FOR END USERS



Fetch credentials

Set up grants

Signed requests

Enables read access for IAM users or corporate director users

Longer-term delegation than presigned URLs allow

Scales to millions of grants, with built-in support in the AWS SDKs

Presigned URLs review

More than just AWS requests signed in advance

Presigned URLs do have valid use cases

AWS security best practices mitigate presigned URL risks

For most access, consider alternatives