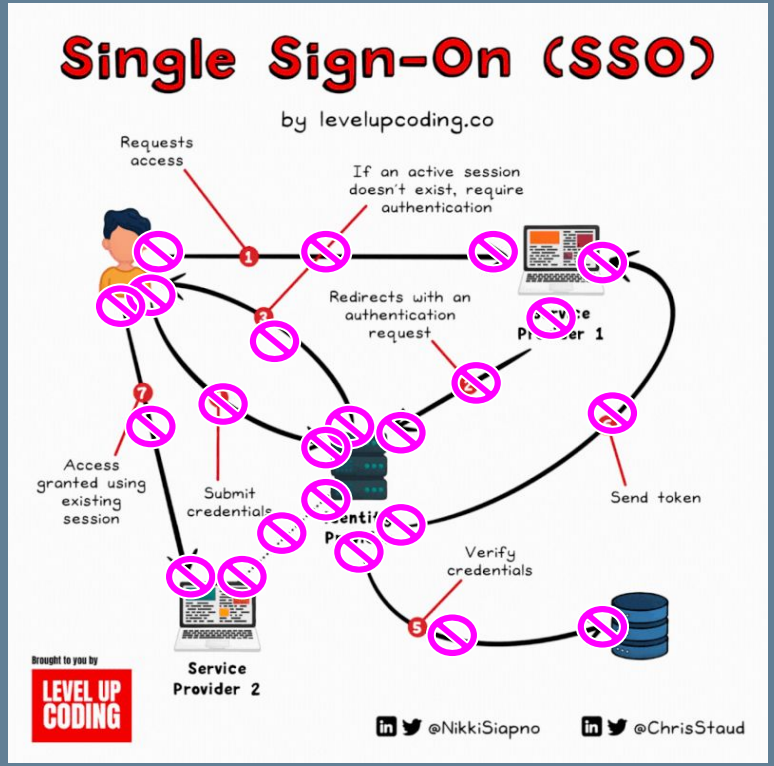
Öryggið í forritunarmálinu


Single Sign-On (SSO)
by levelupcoding.co
Requests access
If an active session doesn't exist, require authentication
Redirects with an authentication request
Access granted using existing session
Submit credentials
Send token
Verify credentials
Service Provider 2
Brought to you by
LEVEL UP CODING
@NikkiSiapno
@ChrisStaud

61% má rekja til lykilorða

Verizon's Data Breach Investigations Report 2023

36% phishing

https://www.techopedia.com/phishing-statistics

Milljarðar af persónulegum gögnum lekið

1. Lykilorð
2. Phishing
3. Skýjaþjónustur

CTO
CEO
Markaðssetning
Bókhald
Þjónustuborð
Gæðaprófanir
Arkitekt
DevOps
Framendi
Bakendi
Db admin
Network admin
Sys admin
Vélbúnaðar uppsetning

Ingi Gauti

26 ár í bransanum
32 ár að forrita

Léttur öryggis nörd
Persónuvernd er mikilvæg

Programming 3.0 - Theory

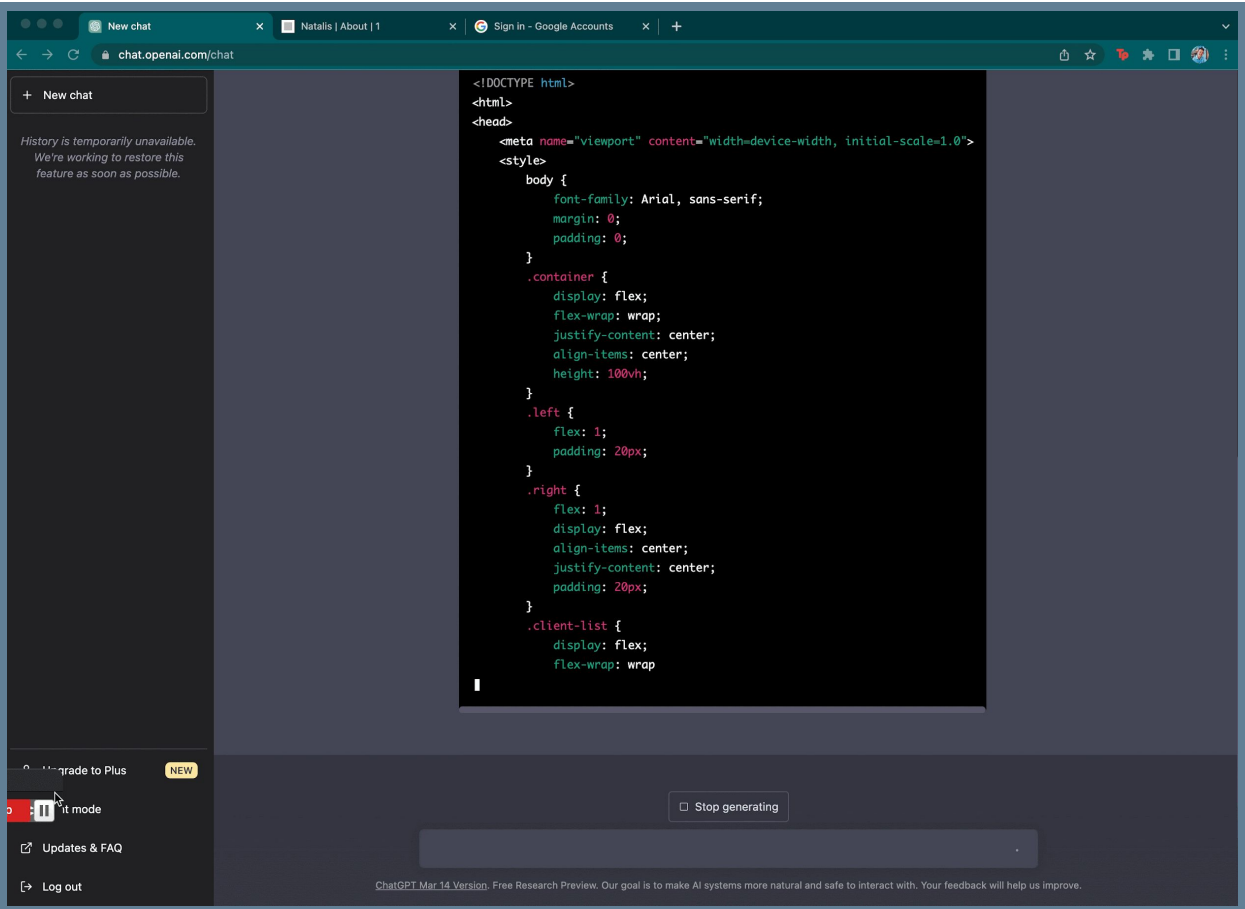
1.5 ár í rannsókn og þróa nýja
tegund forritunarmála

Plang

Hvítbók (white paper)
https://bit.ly/plang_paper

PLang

More at https://plang.is




V.C.C.
1896
MT 90

Hvað er rétta leiðin?

```
{
    “name”:”Ingi”
}
```

Ég heiti Ingi

↓

LLM

↓

```
{
    “name”:”Ingi”
}
```

Náðu í url.com og settu inn í %breytu%

↓

LLM

↓

```
{
    “module”:”Http”
    “method”: “Get”
    “return” : string
}
```

{ “module”:”Http”...

```
public async Task<(object?, IError?)> Request(string url, string method, object? data = null, bool doNotSignRequest = false,
    Dictionary<string, object>? headers = null, string encoding = "utf-8", string contentType = "application/json", int timeoutInSeconds = 30)
{
    var requestUrl = variableHelper.LoadVariables(url);
    if (requestUrl == null)
    {
        return (null, new ProgramError("url cannot be empty", goalStep, function));
    }
    if (!requestUrl.ToString().ToLower().StartsWith("http"))
    {
        requestUrl = "https://" + requestUrl;
    }

    using (var httpClient = httpClientFactory.CreateClient())
    {
        var httpMethod = new HttpMethod(method);
        var request = new HttpRequestMessage(httpMethod, requestUrl.ToString());

        if (headers != null)
        {
            foreach (var header in headers)
            {
                var value = variableHelper.LoadVariables(header.Value);
                if (value != null)
                {
                    request.Headers.TryAddWithoutValidation(header.Key, value.ToString());
                }
```

LLM er ekki forrita

LLM strúktúrar meiningu yfir á núverandi kóða

Allar HTTP fyrirspurnir í tungumálinu fara í gegnum þennan kóða

Tækifæri

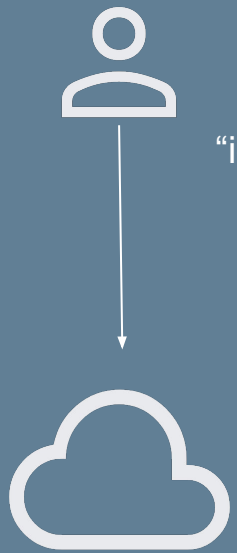
Við getum nýtt 60 ára reynslu af öryggi

Og sett í þennan kóða

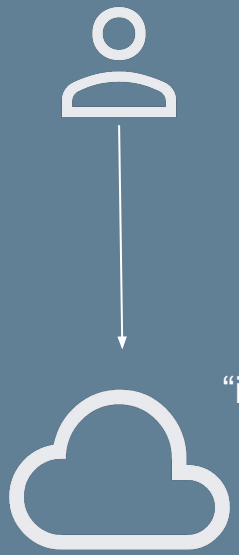
Undirskrift á allar fyrirspurnir

Signing

```
{
    “created”:”2024-11-08T23:...”
    “nounce”: “1234-4567…”
    “body”: “abcdeft….”
    “identity”: “abcdef123…”
    “signature”: “jklweiuosjlkw…”
}
```



“identity”: “abcdef123…”



“identity”: “abcdef123…”

Innskráning

```
select * from users where
email='ingi@plang.is' and password='123456'
```

Skráning / Innskráning / Gleymt lykilorð

1. ~~Lykilorð~~
2. Phishing
3. Skýjaþjónustur

Phishing

Öll skilaboð eru undirrituð

Því veit ég að skilaboðin
eru að koma frá
ákveðnum aðila

1. ~~Lykilorð~~
2. ~~Phishing~~
3. Skýjaþjónustur

Skýjaþjónustur

Af hverju?

Leysir deilingu gagna

Skýjaþjónustur

Settu “Skrifa undir $5 milljarða samning við Microsoft” í Task listann → LLM ↓

```
{
    “module”:”Database”
    “method”: “Insert”
    “return” : string
}
```

{ “module”:”Database”...

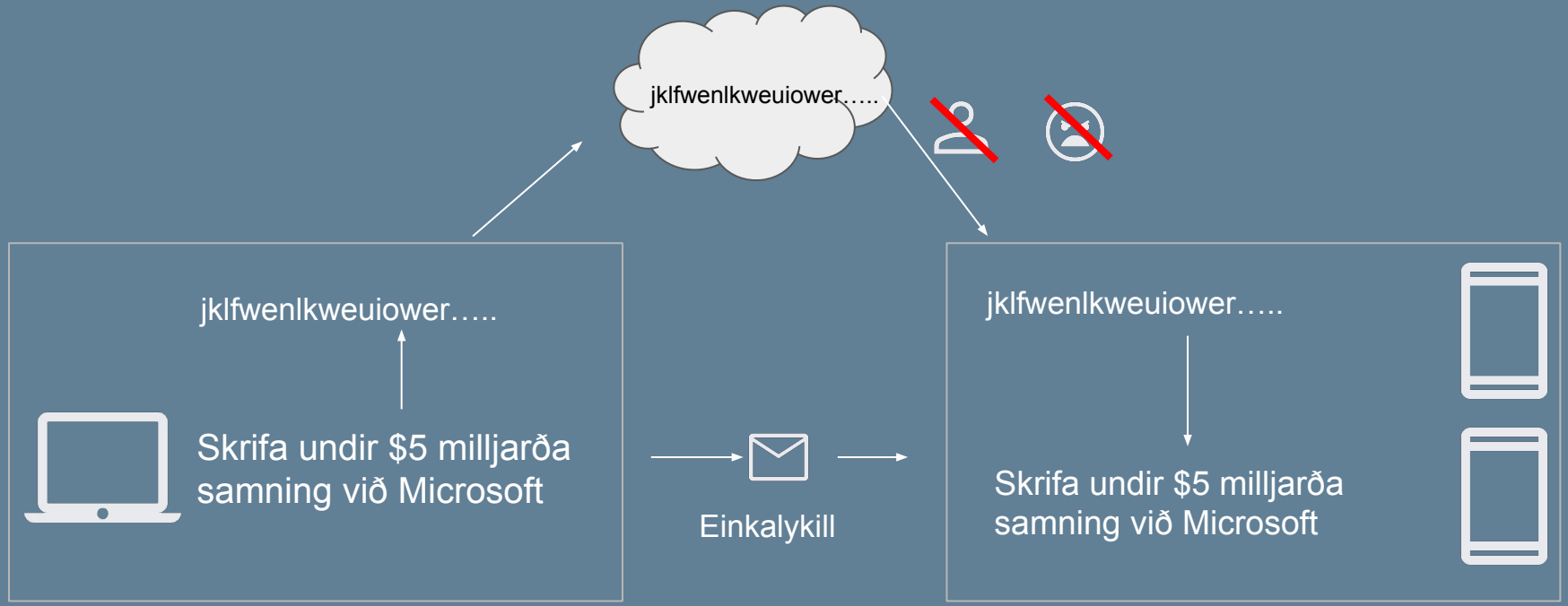
```
[Description("Basic insert statement. Will return affected row count")]
3 references | 0/2 passing
public async Task<(int rowsAffected, IError? error)> Insert(string sql, List<object>? SqlParameters = null, string? dataSourceName = null)
{
    if (!string.IsNullOrEmpty(dataSourceName))
    {
        await SetDataSourceName(dataSourceName);
    }
    int rowsAffected = 0;
    var prepare = Prepare(sql, SqlParameters, true);
    if (prepare.error != null)
    {
        return (0, prepare.error);
    }
    try
    {

        if (eventSourceRepository.GetType() != typeof(DisableEventSourceRepository))
        {
            rowsAffected = await eventSourceRepository.Add(prepare.connection, prepare.sql, prepare.param);
        }
        else
        {
            rowsAffected = await prepare.connection.ExecuteAsync(prepare.sql, prepare.param);
        }
    }
    catch (Exception ex)
    {
        if (GoalHelper.IsSetup(goalStep) && ex.ToString().Contains("duplicate key"))
        {
            ShowWarning(ex);
            return (rowsAffected, null);
        }
        return (0, new ProgramError(ex.Message, goalStep, function, Exception: ex));
    }
    finally
    {
        Done(prepare.connection);
    }
    return (rowsAffected, null);

}
```

Allar gagnagrunns
fyrirspurnir í tungumálinu
fara í gegnum þennan kóða

Tækifæri

Dulkóðum og undirbúum fyrir flutning

Skýjaþjónustur



https://ingig.substack.com/p/plang-and-local-first

1. ~~Lykilorð~~
2. ~~Phishing~~
3. ✅Skýjaþjónustur

1

Veikleiki

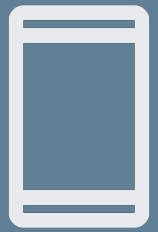
Einkalykillinn

Private key

Single point of failure

Gott í þessu tilfelli

Við getum sett allan fókus á þennan eina veika punkt



Símar eru frábær lausn fyrir einkalykla

Alltaf hægt að bæta við öryggislagi

MFA Multi sig

~~Lykilorð~~

~~Phishing~~

✅Skýjageymslur

1 veikleiki - verndaður

Annað

90%+ minni kóði => færri villur, færri öryggisvandamál
Undirritaður & gegnsær kóði

Framtíðin

Sjálfvirkar prófanir
Sjálfleiðréttandi hugbúnaður
Ónafngreint KYC
Isolated User Storage Pattern

https://ingig.substack.com/

Plang

Útgáfa 0.15.5

Brjótandi breytingar
Fullt af öryggis göllum
Ennþá að finna tækifærin

Open source (LGPL 2.1)
plang.is | github.com/PLangHQ

Styrkja verkefnið?

Plang will become the most secure programming language, in any sense.

Plang will give its users more privacy than any development tool can give, in any sense.

plang.is | github.com/PLangHQ