

Part C: Omnihash & Omni*Shell

- Hash of Everything
- “Less is More”, “keep it simple”.

All kinds of digital assets, including personal information, text, photos, videos, files, contracts. program source code.

- Trispecies Monetary Payments: Fiat Currency, Cryptocurrency, Bullions (Gold/Silver)

Omni = anything / everything. Opposite of UNIX.

* = Everything in UNIX shells, avoid legal conflicts.

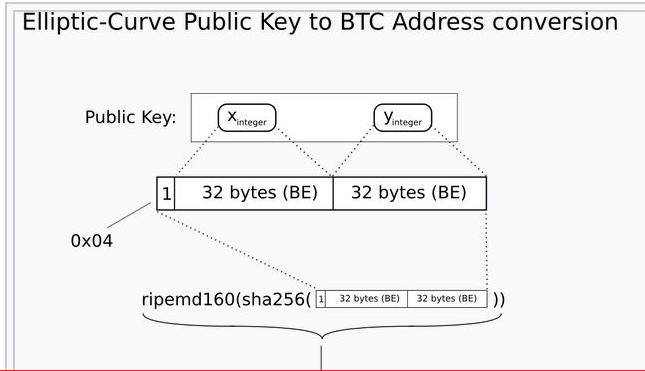
https://en.bitcoin.it/wiki/Technical_background_of_version_1_Bitcoin_addresses



Technical background of version 1 Bitcoin addresses

This article may be too technical for some users. The more basic article on [Bitcoin Addresses](#) may be more appropriate.

A Bitcoin address is a 160-bit hash of the public portion of a public/private ECDSA keypair. Using public-key cryptography, you can "sign" data with your private key and anyone who knows your public key can verify that the signature is valid.



Hardest: finding the simplest & most fundamental definition & replicate in a new framework – luck & persistence

[Printable version](#)

[Permanent link](#)

[Page information](#)

[Sister projects](#)

associated private keys (or the seed needed to generate them) are stored in the [wallet](#) data file.

This is the only file users should need to [backup](#). A "send" transaction to a specific Bitcoin address

Omni*Web: Def'n Ownership of Digital Assets w' Ownership JSON, bs'd on decentral...

https://8a75-2001-d08-da-5986-f093-ef4e-d34-4828.ngrok-free.app//oxw/oxw.php?nn=adam

150%

Hash of public key as a Decentralised User Identifier

<https://www.youtube.com/watch?v=FofGUYJS7js>

3. Enter "myhash e:" and click on the Nickname button labeled "adams"

The server will return the hash of your pulic key

4. Press F12 to open the Developer Tools, click

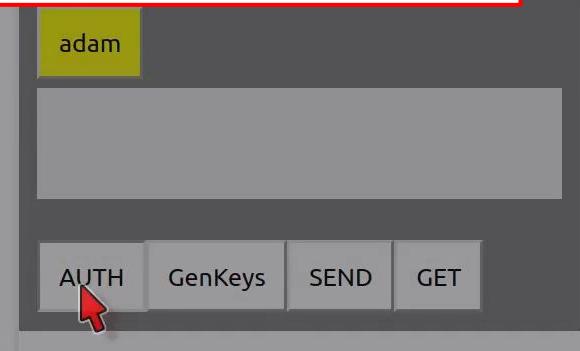
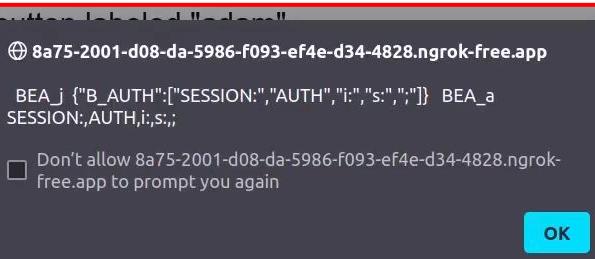
Enter the following command to adjust the position

```
document.querySelectorAll('.chat-popup')[1].style
```

```
document.querySelectorAll('.chat-popup')[1].style.right = -400px
```

```
document.querySelectorAll('.chat-popup')[1].style.display="none"
```

```
document.querySelectorAll('.chat-popup')[1].style.display="block"
```

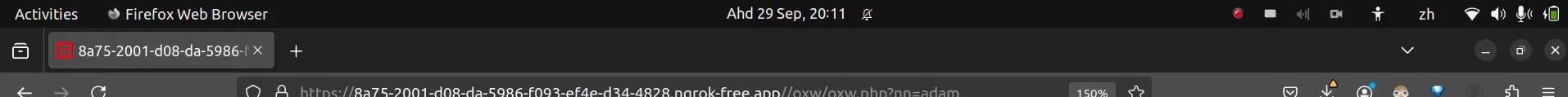


```
fgl_s 333 < 2 > array ( 0 => array ( 0 => '{"B_AUTH": ["SESSION:", "AUTH", "i:", "s:", ";"]}', 1 => 'jd:', 2 => 'cdw:', 3 => 'am:', 4 => 'lcdw:', 5 => 'B_AUTH', ), 1 => NULL, )
```



4:01 / 9:36





Omni*Web Login Demo 2024-08

omnixtar.github.io Home Page

1. Click AUTH button to initiate Authentication.

Server will return Public Key PBK = null if private-public keypair is not initialised.

2. Click GenKeys to generate private-public keypair. Then click AUTH again to initiate Authentication.

Server will return Public Key PBK and browser will display it.

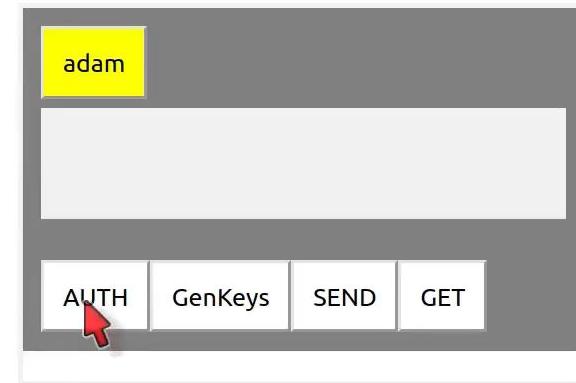
3. Enter "myhash e:" and click on the Nickname button labeled "adam".

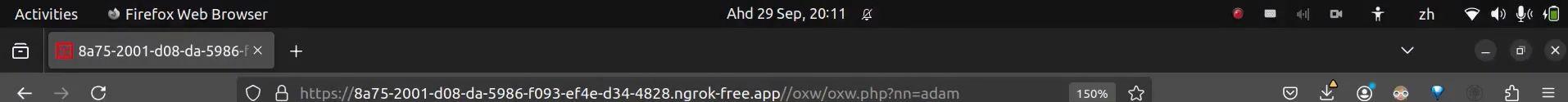
The server will return the hash of your public key (HPBK).

4. Press F12 to open the Developer Tools, click Console tab to open browser console.

Enter the following command to adjust the position and visibility of the dialog box:

```
document.querySelectorAll('.chat-popup')[1].style.bottom="200pt"
document.querySelectorAll('.chat-popup')[1].style.right="400pt"
document.querySelectorAll('.chat-popup')[1].style.display="none"
document.querySelectorAll('.chat-popup')[1].style.display="block"
```





Omni*Web Login Demo 2024-08

omnixtar.github.io Home Page

1. Click AUTH button to initiate Authentication.

Server will return Public Key PBK = null if private-public keypair is not initialised.

2. Click GenKeys to generate private-public keypair. Then click AUTH again to initiate Authentication.

Server will return Public Key PBK and browser will display it.

3. Enter "myhash e:" and click on the Nickname button labeled "adam".

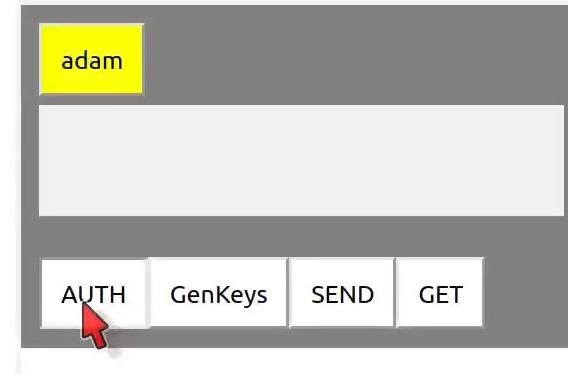
The server will return the hash of your pulic key (HPBK).

4. Press F12 to open the Developer Tools, click Console tab to open browser console.

Enter the following command to adjust the position and visibility of the dialog box:

```
document.querySelectorAll('.chat-popup')[1].style.bottom="200pt"
document.querySelectorAll('.chat-popup')[1].style.right="400pt"
document.querySelectorAll('.chat-popup')[1].style.display="none"
document.querySelectorAll('.chat-popup')[1].style.display="block"
```

```
fgl_s 333 < 2 > array ( 0 => array ( 0 => '{"B_AUTH":["SESSION:","AUTH","i:","s:","",""]}', 1 => 'jd:', 2 => 'cdw:', 3 => 'am:', 4 => 'lcdw:', 5 => 'B_AUTH', ), 1 => NULL, )
```



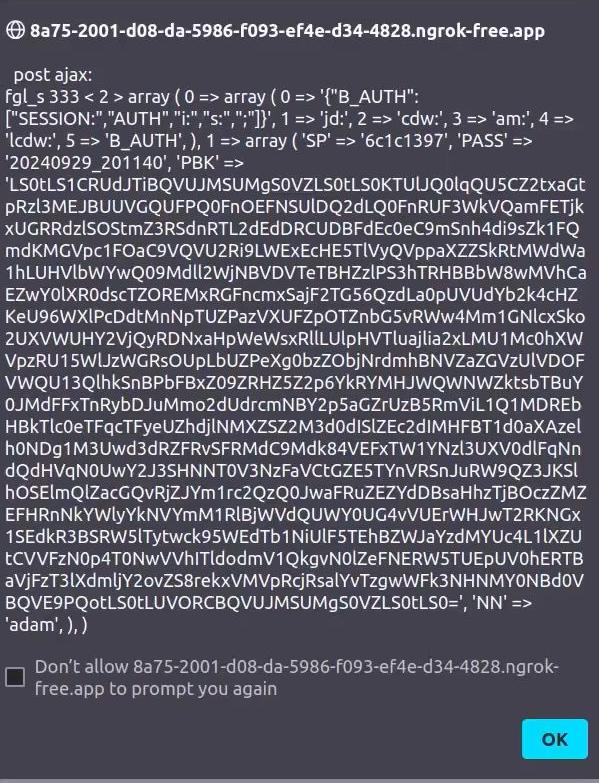
Omni*Web Login Demo 2024-08

omnixtar.github.io Home Page

1. Click AUTH button to initiate Authentication.
Server will return Public Key PBK = null if private key is not generated.
2. Click GenKeys to generate private-public key pair.
Server will return Public Key PBK and browser will prompt for download.
3. Enter "myhash e:" and click on the Nickname
The server will return the hash of your public key.
4. Press F12 to open the Developer Tools, click on the Network tab to view the request and response.

```
document.querySelectorAll('.chat-popup')[1].style.display = 'block';  
document.querySelectorAll('.chat-popup')[1].style.height = 'auto';  
document.querySelectorAll('.chat-popup')[1].style.overflow = 'auto';  
document.querySelectorAll('.chat-popup')[1].style.whiteSpace = 'pre';
```

```
fgl_s 333 < 2 > array ( 0 => array ( 0 => '{"B_AUTH": "SESSION", "AUTH": "POST", "s": "", "t": ""}, 1 => 'jd:', 2 => 'cdw:', 3 => 'am:', 4 => 'lcdw:', 5 => 'B_AUTH', ), 1 => array ( 'SP' => '6c1c1397', 'PASS' => '20240929_201140', 'PBK' =>
```



cation.



```
':, 2 => 'cdw:', 3 => 'am:', 4 => 'lcdw:', 5 =>
```

Don't allow 8a75-2001-d08-da-5986-f093-ef4e-d34-4828.ngrok-free.app to prompt you again

OK

8a75-2001-d08-da-5986-f × +
https://8a75-2001-d08-da-5986-f093-ef4e-d34-4828.ngrok-free.app//oxw/oxw.php?nn=adam 150% ☆
Home Page

1. Click AUTH button to initiate Authentication.

Server will return Public Key PBK = null if private-public keypair is not initialised.

2. Click GenKeys to generate private-public keypair. Then click AUTH again to initiate Authentication.

Server will return Public Key PBK and browser will display it.

3. Enter "myhash e:" and click on the Nickname button labeled "adam".

The server will return the hash of your public key (HPBK).

4. Press F12 to open the Developer Tools, click Console tab to open browser console.

Enter the following command to adjust the position and visibility of the dialog box:

```
document.querySelectorAll('.chat-popup')[1].style.bottom="200pt"
document.querySelectorAll('.chat-popup')[1].style.right="400pt"
document.querySelectorAll('.chat-popup')[1].style.display="none"
document.querySelectorAll('.chat-popup')[1].style.display="block"
```

```
fgl_s 333 < 2 > array ( 0 => array ( 0 => '{"B_AUTH":["SESSION:","AUTH","i:","s:","",""]}', 1 => 'jd:', 2 => 'cdw:', 3 => 'am:', 4 => 'lcdw:', 5 => 'B_AUTH', ), 1 => NULL, )
fgl_s 333 < 2 > array ( 0 => array ( 0 => '{"B_AUTH":["SESSION:","AUTH","i:","s:","",""]}', 1 => 'jd:', 2 => 'cdw:', 3 => 'am:', 4 => 'lcdw:', 5 => 'B_AUTH', ), 1 => array ( 'SP' => '6c1c1397', 'PASS' => '20240929_201140', 'PBK' =>
'LS0tLS1CRUdJTiBQVUJMSUMgS0VZLS0tLS0KTUIJQ0lqQU5CZ2txaGtpRzl3MEJBUVGQUFPQ0FnOEFNSUIDQ2dLQ0FnRUF3WkVQamF
'NN' => 'adam', ), )
fgl_s 333 < 2 > array ( 0 => array ( 0 => '{"B_AUTH1":["SESSION:","AUTH","i:","PBK","i:","o_hash.json","l_cdwjs","h_b64","s:","",""]}', 1 => 'jd:', 2
=> 'cdw:', 3 => 'am:', 4 => 'lcdw:', 5 => 'B_AUTH1', ), 1 => 'Graph/dmeta/HnzRI9nOdQ==', )
```



Omni*Web: Def'n Ownership of Digital Assets w/ Ownership JSON, bs'd on decentral...



```
5 // commands to create Ownership JSON
6 jb={owner: "HnzRl9n0dQ=="}
7 jb.nick='adam'
8 jb.jstr='{"phone":123,"email":"bac@abc","name":"beta"}'
9 jb.hjs=bnToB64(cyrb53(jb.jstr))
10
11 // Ownership JSON object in console
12 jb
13 Object {
14   owner: "HnzRl9n0dQ==",
15   nick: "adam",
16   jstr: '{"phone":123,"email":"bac@abc","name":"beta"}' ,    [
17   hjs: "GMpBTscsMw=="
18 }
19
20 // JSON string ownership JSON
21 JSON.stringify(jb)
22 '{"owner": "HnzRl9n0dQ==", "jstr": {"\\\"phone\\\":123, \\\"email\\\":\\\"bac@abc\\\", \\\"name\\\":\\\"beta\\\""}, "hjs": "GMpBTscsMw=="}'
23
24 // JSON string ownership JSON, prettified
25 [
26   "owner": "HnzRl9n0dQ==",
27   "jstr": {"\\\"phone\\\":123, \\\"email\\\":\\\"bac@abc\\\", \\\"name\\\":\\\"beta\\\""}, "hjs": "GMpBTscsMw=="
28 ]
```



1:26 / 9:36



Console Network Style Editor Performance »

Filter Output Errors Warnings Logs Info Debug CSS XHR Requests

```
» jd.jstr="yo6sgm7q/pfvvp2e4kcuhjtf7wfltt63igwcukhbmum6lu3a3a.b32.i2p"
← "yo6sgmfq7pfvvp2e4kcuhjtf7wfltt63igwcukhbmum6lu3a3a.b32.i2p"

» jd
← ▶ Object { owner: "LlwHci_t", nick: "donald", jstr:
  "yo6sgmfq7pfvvp2e4kcuhjtf7wfltt63igwcukhbmum6lu3a3a.b32.i2p", date: "2024-10-18T15:18:49.260Z", hjs:
  "Bw7BA+t48Q==" }

» hjs=bnToB64(cyrb53(jd.jstr))
← "A0t0bR270w=="

» jd
← ▼ Object { owner: "LlwHci_t", nick: "donald", jstr:
  "yo6sgmfq7pfvvp2e4kcuhjtf7wfltt63igwcukhbmum6lu3a3a.b32.i2p", date: "2024-10-18T15:18:49.260Z", hjs:
  "Bw7BA+t48Q==" }
  date: "2024-10-18T15:18:49.260Z"
  hjs: "Bw7BA+t48Q=="
  jstr: "yo6sgmfq7pfvvp2e4kcuhjtf7wfltt63igwcukhbmum6lu3a3a.b32.i2p"
  nick: "donald"
  owner: "LlwHci_t"
  ▶ <prototype>: Object { ... }
```

Inspector Debugger Console Network Style Editor Performance »

Filter Output Errors Warnings Logs Info Debug CSS XHR Requests

```
» cyrb53.toString()
← "function(str, seed = 0) {
    let h1 = 0xdeadbeef ^ seed, h2 = 0x41c6ce57 ^ seed;
    for (let i = 0, ch; i < str.length; i++) {
        ch = str.charCodeAt(i);
        h1 = Math.imul(h1 ^ ch, 2654435761);
        h2 = Math.imul(h2 ^ ch, 1597334677);
    }
    h1 = Math.imul(h1 ^ (h1>>>16), 2246822507) ^ Math.imul(h2 ^ (h2>>>13), 3266489909);
    h2 = Math.imul(h2 ^ (h2>>>16), 2246822507) ^ Math.imul(h1 ^ (h1>>>13), 3266489909);
    return 4294967296 * (2097151 & h2) + (h1>>>0);
}"
```

```
» bnToB64.toString()
← "function bnToB64(bn) {
    var hex = BigInt(bn).toString(16);
    if (hex.length % 2) { hex = '0' + hex; }

    var bin = [];
    var i = 0;
```

Inspector Debugger Console Network Style Editor Performance »

Filter Output Errors Warnings Logs Info Debug CSS XHR Requests

```
    return 4294967296 * (2097151 & h2) + (h1>>>0);  
}"
```

```
» bnToB64.toString()
```

```
← "function bnToB64.bn) {  
  var hex = BigInt.bn).toString(16);  
  if (hex.length % 2) { hex = '0' + hex; }  
  
  var bin = [];  
  var i = 0;  
  var d;  
  var b;  
  while (i < hex.length) {  
    d = parseInt(hex.slice(i, i + 2), 16);  
    b = String.fromCharCode(d);  
    bin.push(b);  
    i += 2;  
  }  
  
  return btoa(bin.join(''));  
}"
```

Part C: Omni*Shell

Omni*Shell: FORTH-shell in JavaScript

Phoscript refers to the programming language, which is a subset of FORTH.

Reason for using Omni*Shell? After quick demo.

Inspector Debugger Console Network Style Editor Performance » ⓘ 1

Filter Output Errors Warnings Logs Info Debug CSS XHR Requests ⚙

```
11 d
1176 before FGLA [[ "d", 11, {} ]]
0 e
1 f
1183 after FGLA []

← undefined
» f('3 4 +')
3 4 +
0 3
1 4
2 +
← undefined
» s
← ▶ Array(9) [ "a", "b", "c", "g", "h", "i", "e", "f", 7 ]
» omnistart()
```

Simple FORTH style addition. Stack s.

Inspector Debugger Console Network Style Editor Performance » 1

Filter Output Errors Warnings Logs Info Debug CSS XHR Requests

omni.js:75:19

2 + ← undefined

» s ← ▶ Array(9) ["a", "b", "c", "g", "h", "i", "e", "f", 7]

» f_add.toString() ← "function(){s.push(parseInt(s.pop())+parseInt(s.pop()))}"

» f('44 55 add:')

44 55 add: ← undefined

0 44

1 55

2 add:

Word form add: → f_add()

omni.js:129:17

omni.js:75:19

omni.js:75:19

omni.js:75:19

» undefined

» s ← ▶ Array(10) ["a", "b", "c", "g", "h", "i", "e", "f", 7, 99]

» Top

Inspector Debugger Console Network Style Editor Performance » ⓘ 1

Filter Output Errors Warnings Logs Info Debug CSS XHR Requests

```
"Bw7BA+t48Q==" }
```

```
» jd
← ► Object { owner: "LlwHci_t", nick: "donald", jstr:
  "yo6sgmfq7pfvvp2e4kcuhtfg7wfllt63igwcukhbmqm6lu3a3a.b32.i2p", date: "2024-10-18T15:18:49.260Z", hjs:
  "Bw7BA+t48Q==" }

» f_add.toString()
← "function(){s.push(parseInt(s.pop())+parseInt(s.pop()))}"

» f_h5364=function(){s.push(bnToB64(cyrb53(JSON.stringify(s.pop()))))}

← ► function f_h5364()

» s
← ► Array(10) [ "a", "b", "c", "g", "h", "i", "e", "f", 7, 99 ]

» jd
← ► Object { owner: "LlwHci_t", nick: "donald", jstr:
  "yo6sgmfq7pfvvp2e4kcuhtfg7wfllt63igwcukhbmqm6lu3a3a.b32.i2p", date: "2024-10-18T15:18:49.260Z", hjs:
  "Bw7BA+t48Q==" }

» s.push(id)
»
```

Inspector Debugger Console Network Style Editor Performance » 1

Filter Output Errors Warnings Logs Info Debug CSS XHR Requests

```
← ▶ function f_h5364()
» s
← ▶ Array(10) [ "a", "b", "c", "g", "h", "i", "e", "f", 7, 99 ]
» jd
← ▶ Object { owner: "LlwHci_t", nick: "donald", jstr:
  "yo6sgmfq7pfvvp2e4kcuhjtfq7wfltt63igwcukhbmqm6lu3a3a.b32.i2p", date: "2024-10-18T15:18:49.260Z", hjs:
  "Bw7BA+t48Q==" }
» s.push(jd)
← 11
» f('h5364:')
h5364:
  0 h5364:
← undefined
» s[10]
← "HnqGtQrVYg=="
»
```

push, execute f_h5364(), top of stack

omni.js:129:17
omni.js:75:19

Top

Inspector Debugger Console Network Style Editor Performance » ⓘ 1

Filter Output Errors Warnings Logs Info Debug CSS XHR Requests

```
← ▶ Array(10) [ "a", "b", "c", "g", "h", "i", "e", "f", 7, 99 ]
» jd
← ▶ Object { owner: "LlwHci_t", nick: "donald", jstr:
  "yo6sgmfp7pfvvp2e4kcuhjtfq7wfllt63igwcukhbmuqm6lu3a3a.b32.i2p", date: "2024-10-18T15:18:49.260Z", hjs:
  "Bw7BA+t48Q==" }
» s.push(jd)
← 11
» f('h5364:')
h5364:
0 h5364:
← undefined
» s[10]
← "HnqGtQrVYg=="
» bnToB64(cyrb53(JSON.stringify(jd)))
← "HnqGtQrVYg=="

verify f_h5364() – same as pure JavaScript
```

omni.js:129:17
omni.js:75:19

Top

Inspector Debugger Console Network Style Editor Performance » ⓘ 1

Filter Output Errors Warnings Logs Info Debug CSS XHR Requests

```
» f_b64=function(){s.push(bnToB64(s.pop()))}
← ► function f_b64()

» f_e=function(){s.push(eval(s.pop()))}
← ► function f_e()

» f('jd e: je: h53: b64:')
    jd e: je: h53: b64:
    0 jd
    1 e:
    2 je:
    3 h53:
    4 b64:
← undefined
» s[12]
← "HnqGtQrVYg=="

»
```

omni.js:129:17

omni.js:75:19

omni.js:75:19

omni.js:75:19

omni.js:75:19

omni.js:75:19

omni.js:75:19

Top

Multiple smaller Phoscript words / JavaScript functions

Inspector Debugger Console Network Style Editor Performance » 1

Filter Output Errors Warnings Logs Info Debug CSS XHR Requests

```
» f_je=function(){s.push(JSON.stringify(s.pop()))}
← ► function f_je()
» f_h53=function(){s.push(cyrb53(s.pop()))}
← ► function f_h53()
» f_b64=function(){s.push(bnToB64(s.pop()))}
← ► function f_b64()
» f_e=function(){s.push(eval(s.pop()))}
← ► function f_e()
» f('jd e: je: h53: b64:')
jd e: je: h53: b64:
0 jd
1 e:
2 je:
3 h53:
```

omni.js:129:17
omni.js:75:19
omni.js:75:19
omni.js:75:19
omni.js:75:19

Multiple smaller Phoscript words / JavaScript functions

Bidirectional Shunting Yard Algorithm (BiSYA)

<https://youtu.be/mYjKS0KijVg>

Infix: $3 + 4$
C, C++
PHP, Python
Java,
JavaScript
.....

Shunting Yard Algorithm (SYA)

Inverse Shunting Yard Algorithm (ISYA)

Reverse Polish Notation
 $3\ 4\ +$
Forth
Phoscript
 $\Phi\Omega\Sigma$



main ▾

omnixtar.github.io / oxshell / js / omni.js

↑ Top

Code

Blame

<https://github.com/omnixtar/omnixtar.github.io/blob/main/oxshell/js/omni.js>

```
58  ▼  phosinit=function(){
59  ▼  Phos=function(){
60      // var S=[] // this is local, not accessible outside
61      $ = this // macro
62      this.S=[];
63      var S=this.S // still need var S for local code access
64      S[0]=[]
65      var S0=S[0]
66      var $CDW = {}
67      S[0].$CDW = $CDW;
68      S0.skip = 0;
69      S0.CDW = [];
70      S0.dlb = {};
71  ▼  var FGLA = function($WA) {
72      // arguments[0].split(' ').map(e=>{
73      var c_cdw=false; var i=0, ic, w=$WA;
```

i2p.i2p/core/java/src/net

omnixtar.github.io/oxshell

+

github.com/omnixtar/omnixtar.github.io/blob/main/oxshell/js/omni.js

Finish update



main ▾

omnixtar.github.io / oxshell / js / omni.js

↑ Top

Code

Blame

Raw



```
58     phosinit=function(){
59         S0.CDW = [];
60         S0.dlb = {};
61     var FGLA = function($WA) {
62         // arguments[0].split(' ').map(e=>{
63         var c_cdw=false; var i=0, ic, W=$WA;
64         $WA.map(e=>{
65             console.log(i, e);
66             var $v=e, $vk=i; $l=$v.length;
67             if (!c_cdw && $v==':') {
68                 c_cdw=true;
69                 console.log(' CDW start ', W[i+1])
70                 ic = i+2; // start index of CDW
71                 CDN=W[i+1]
72                 $CDW[CDN]=[]
73             }
74         }
75     }
```



main ▾

omnixtar.github.io / oxshell / js / omni.js

↑ Top

Code

Blame

Raw



```
58     phosinit=function(){
71         var FGLA = function($WA) {
110             S0.CDW.pop();
111             console.log(1183, 'after FGLA', JSON.stringify(S0.CDW));
112             S0.cda = end(S0.CDW);
113         }
114         else if ($v[$l - 1] == ':') { // colon suffix word after symbol else : will fail
115             var $fn = $v.substr(0, $l-1);
116             if (typeof eval("f_"+$fn)!=="undefined")
117                 // console.log('is func', $v, typeof eval("f_"+$fn))
118                 eval("f_"+$fn+"()")
119             } else s.push(e)
120         }
121         i++
122     }
123 }
```

 main	omnixtar.github.io / oxshell / js /	 Top
 custom.js	Add files via upload	2 months ago
 gz-omni.js	Add files via upload	2 months ago
 jquery-3.4.1.min.js	Add files via upload	2 months ago
 omni.js	Add files via upload	2 months ago
 omni.js.gz	Add files via upload	2 months ago
 omnigz.js	Add files via upload	2 months ago
 s0.js	Add files via upload	2 months ago
 sz0.js	Add files via upload	2 months ago

raw.githubusercontent.com/omnixtar/omnixtar.github.io/refs/heads/main/oxshell/js/omnigz.js

```
gunzipinit=function(){gunzip=e=>{const r=Uint8Array.from(atob(e),(e=>e.charCodeAt(0))),n=new DecompressionStream("gzip"),t=n.writable.getWriter();return t.write(r),t.close(),new Response(n.readable).arrayBuffer().then((function(e){return(new TextDecoder).decode(e)})),getgzfn_js=async function(s){var s1=await gunzip(s),j3=JSON.parse(s1),j4=JSON.parse(j3.s);return eval(j4.n+="+"+j4.s),j3}}}  
omnigz="H4sIAAAAAAAA8UZa1PbSPIz/hUdFxdrYqFY5o1PbBEI2aQg2Ttyy255vdRYGtksuSSxjzC8t+ve2YkzbkNp/0H0Dq6e7pd/eM4lkaKpmlINMwmUXCmeUJg8cGwC3PoYAAoiycTUSqvDAXXIn3iaA3p1mEuZyqJushbuEVQh0plcvhTAmnqR6mouICU4l79faa3/LvIRd5iLi0L61W2w2z6MHj06lIo+0xTCKnwPWhRuPtW5jciFueXCV0M5ukcsyLcZMR/GTy8WP3g5/c3QVBo1wKYquiwx4b4UM+3N6cgwqVu1AIeGiHUZqAyRCwdgnwH0keDQUIoY/NZIL466Gb/nhTi2dy28p+niLAeHiCXhuBC0e/j4T8A9vESkI0Xv7baxrvmFY0Sl9XDM8+MsEkfKkaxXrWsxzrkae3IySxx8/RNpx0jubG9tbW7v7vh1504ictci+9v7u5ubWzu7uxb5qfESb/x7eHjo7zDcoru1s9ftbnd2GS4sscW/hLeJeJvdnZ2tvf39zr7l/pwYFv+7fGv7P8s3F2qWp7DV3d/a39nt7u/AG3C6nf1df9uH17gtg7Zl0NGRMky/Zu92tipngwU4w5SVAT4W9yjt0zn6iCGNcE9lFxiY6chBYwlGSNPcW8dCP8A30bRkrU6LdwSn3vapMRvKFNC6A969l2HQvkSlQ9DerjDoBbgUITMNyidg+JxyvNCkFy0XCQyFI50kwMbZUCv7li7DBHVy0zFeTY5tpHkROW6TL3prBg7QwtADgF06ZmkTMYdqow7hHqdydRptZg2YZlo84Q5/cweAVPtzVUCE8HTApIsHVHsT2Ao8J+AiUx1LEVklb32g8f04PQzbXbte0VALB18qlubmUUVOkm4gx0sMw6j5Y8XX2oYml03eCw0Pl18+ewVekXGD8iNue0D0r8myZ1VHaMzmGWFlkivCQb0UTEp5/bzaDZjkERfw6fGhztgmWI6s0SbdFhugw1P5zsHclf99NoFARNWkxyo1n6TU7RRMrUrJ9/u3t7cx7df0PcxnyxXrUMNBDB4SNJriAP/iNTtXeU5/xB093hKhs6grk0Iol6NUGRmJsG2qoizCbTXBQFskWrCj5BcZBzk7kqSL27XCo+RL0MhLrEZ5E7rGfVV3pV0DmiemGSFegfl5j+WxRT1Ek4qYcMI6JnHifB3s3imFh4aixSx6n0EezRMNWu/oodggSLRM68SD8gxhNjTy6KMfoWp1fxRcCLhzSEedVmj7o7+QG/41KBMRCC3evNQLtd545TYFhcb9Uh15vo11IpE4hbXmpcu+VpBk/LPQa7VajusSfNjMfeD8/3Pv32aaT+Fwb3FMPZaCRyzQxI/IPNzQPfb8yp6q4ksYwb9m4zGUEnCII7mUbZnVe8fu2Uj0F/wHqEqKrVxtUVeRS9v8WHM1kokYr8pxdXnCYGCzVgLbk7d0DGhnikcmzhv2ez1q0ATBFsgCqDDJstwCogMmi2M3S3RLa/tZu1t9+ZK4Jqa2Fmg1OMxF+wfcinInJrBMqUIUhX0EXUKUqggpgK5Ci/DSSD/+ev5GXPrYdksp2mmmyKFrY+HHTRU86AZBGm/hG34g5/OvVMnZQfCk2lqGAVILXkY7mSTExHzWaIcDCz3lc805kZTiodjbTccJtIXLPb3DYapZH2m9/VMpP3Kk5kIXvkU1o3GdJwVyyne+AwBCwCq1BhsFAcXGBH0rMayAEkVN+SJC2mmqIch5TNmHaAJCxkJTbi0PUFjUyDzMM80lCAeMdNvhr0BEV6hZJJASr0QX
```

Omni*Shell in Java

```

2292
2293     logMessage(
2294         Stack<Object> stackAny = new Stack<>();
2295         Stack<HashMap<String, String>> stackHM = new Stack<>();
2296         // Stack<> stackAll = new Stack<>();
2297         String expr = msg;
2298
2299         System.out.println(expr);
2300         System.out.println("Input\tOperation\tStack after");
2301
2302         // Phoscript (Phos) Engine
2303         for (String token : expr.split("\\s+")) {
2304             System.out.print(token + "\t");
2305             switch (token) {
2306                 case "+":
2307                     System.out.print("Operate\t\t");
2308                     stack.push(stack.pop() + stack.pop());
2309                     break;
2310                 case "-":
2311                     System.out.print("Operate\t\t");
2312             }
2313         }
2314     }
2315
2316     System.out.println("Output\tStack after");
2317 }

```

The secret of Shell was given in the 6 Metaprogramming Magic words presented earlier “String Split Array, Loop Push Eval”, and the code is within logMessage() function in BuddyPluginBeta.java, from lines 2321 to 2430. (URL given in YouTube descriptions.)



master ▾

omnixchat / core / src / com / biglybt / plugin / net / buddy / BuddyPluginBeta.java

↑ Top

Code

Blame

<https://github.com/omnixtar/omnixchat/blob/master/core/src/com/biglybt/plugin/net/buddy/BuddyPluginBeta.java>

```
2292         logMessage(  
2293             Stack<Double> stack      = new Stack<>();  
2294             Stack<String> stackStr = new Stack<>();  
2295             Stack<Object> stackAny = new Stack<>();  
2296             Stack<HashMap<String, String>> stackHM = new Stack<>();  
2297             // Stack<> stackAll = new Stack<>();  
2298             String expr = msg;  
2299  
2300             System.out.println(expr);  
2301             System.out.println("Input\tOperation\tStack after");  
2302  
2303             // Phoscript (Phos) Engine  
2304             for (String token : expr.split("\s+")) {  
2305                 System.out.print(token + "\t");  
2306                 switch (token) {  
2307                     case "+":  
2308                         System.out.print("Operate\t\t");  
2309                         stack.push(stack.pop() + stack.pop());  
2310                     break;
```

Super-program or Super-Shell?

Understanding of program – change after using “super shell” – metaprogramming in Phoscript (FORTH).

Interaction between programs, via “super-shell” – become “super-program”.

UNIX pipe is limited to one computer.

Omnihash data + I2P = global data sharing.

Small size of Omni*Shell, 3kb minified gzipped JavaScript, easily portable to other programming languages. (Examples later.)

Strategy: *Trojan Horse Super-Shell*, not the computer virus type – *security guaranteed by hash & public key cryptography*.

Part C: Other Applications of Omnihash

- Hyper hash graph is a graph made up of Omnihash JSON nodes, and is a generalised form of Distributed Hash Table (DHT).
- The simplest and most fundamental smart contract: Omni*Contract based on Omnihash.

Omni*Web: Omni*Contract

OmniHash: Base64 53-bit Hash

<https://omnixtar.github.io/omnicontract/>

Contract to hash:

Brian will transfer USD50 to Trinity, via local banks, after Adam sends TRX-USDT50 to Brian.

OmniHash

OXC3 Triplets:

:Contract_Hash: :PBKA_Hash: :PBKB_Hash:

OXC3

ROOT

OmniHash of Contract:

:Contract_Hash:

Root Hash (Hash of All Hashes):

:Root_Hash:

Omni*Contract (OXC) is arguably the simplest smart contract ever invented.

The simplest form of OXC consists of THREE (3) OmniHashes, which are collectively known as OXC3 or OXC Triplets:

- OmniHash of the Public Key of Party A
- OmniHash of the Public Key of Party B

Activities Firefox Sab 16 Nov, 07:11 zh

Omni*Contract: Decentraliz https://omnitar.github.io/omnicontract/ + view-source:https://omnitar.github.io/omnicontract/ 240% ☆

```
1 <html>
2   <!--
3 Omni*Contract On the Separation of Disclosure and Royalties of the Source Code
4
5 - You, a human agent of a company or government agency, may read the source code
without making payments to the author or authors, but if you execute this
program on behalf of your company or agency for commercial purposes, we reserve
the rights to claim royalties from you or your company or agency.
6
7 -->
8 <head>
9 <title>Omni*Contract: Decentralized Meta Contact</title>
10 <!--[if IE]>
11 <script type='text/javascript'>
12   (function() {
13     var tag = document.getElementById('basetag');
14     tag.href = location.protocol + '//' + location.host + '/jsencrypt/';
15   })();
16 </script>
17 <![endif]-->
18 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
19 <!--[if lt IE 9]>
20   <script src="https://www.w3schools.com/lib/html5shiv.js"></script>
21 <![endif]-->
```



Business Source License

[文](#) [Add languages](#) ▾[Article](#) [Talk](#)[Read](#) [Edit](#) [View history](#) [Tools](#) ▾

From Wikipedia, the free encyclopedia



This article **relies excessively on references to primary sources**. Please improve this article by adding [secondary or tertiary sources](#).

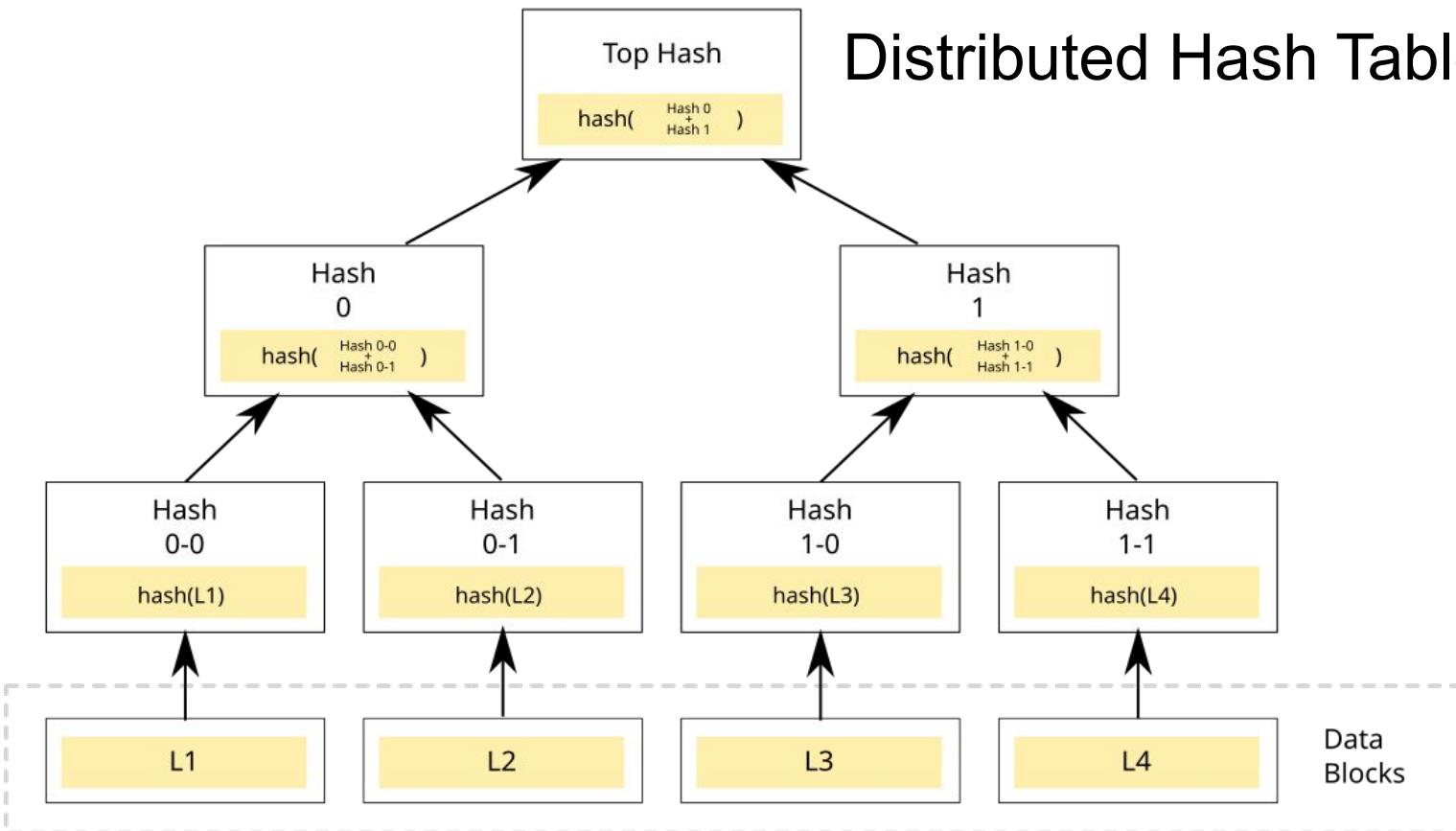
Find sources: "Business Source License" – news · newspapers · books · scholar · JSTOR (January 2024) ([Learn how and when to remove this message](#))

The **Business Source License** (SPDX id [BUSL](#)^[1]) is a [software license](#) which publishes [source code](#) but limits the right to use the [software](#) to certain classes of users. The BUSL is not an [open-source license](#),^[1] but it is [source-available](#) license that also mandates an eventual transition to an open-source license. This characteristic has been described as a compromise between traditional proprietary licenses and open source.^[2]

The originator of the BUSL is [MariaDB Corporation AB](#), where it is used for the MaxScale product, not for the flagship [MariaDB](#).^[3]

[Terms](#) [[edit](#)]

Distributed Hash Table (DHT)



Hyper Hash Graph (HHG)

- *Each node is Omnihash JSON*

