

H-ERP API

המערכת נתמכת בחשבשבת חלונות מגרסת 2020 SP5 ומעלה בלבד.

יש להרשם לשירות בכדי לפתוח את ההרשאה לשימוש במערכת.

למערכת שתי שכבות של אבטחת מידע:

1. הזדהות של החשבשבת המקומית station שהוא טוקן לאימות שרת הלקוח.
2. קובץ אבטחה שחותם כל קריאה וכתובה לשרת המקומי, קובץ זה ניתן להוריד חינם מאתר <https://netpassport.io/register>. קובץ זה חותם את הספק המורשה והנו ייחודי לספק.

הקמת ספק חדש :

ספק חדש היא התוכנה צד ג' המבקשת לכתוב לתוך חשבשבת מקומית באמצעות Rest API. הרישום הינו ברמת הספק ולא ברמת לקוח קצה או ממשק, ז"א נדרש רישום חד פעמי. ניתן להשתמש בהגדרות אלו ללקוחות קצה שונים ולממשקים שונים.

יש להרשם בכתובת (ללא עלות) :

Register into <https://netpassport.io/register> and manage your authenticated users

הגדרות מערכת צד לקוח:

1. יש להכנס ולהגדיר באחת מתחנות חשבשבת. הגדרות << הגדרות H-Connect
 - 1.1. **חברה:** לקליטת הממשק.
 - 1.2. **מזהה הספק:** ה-ID של קובץ האבטחה, הנמצא בשרת הספק והתקבל מה NETPASSPORT.
 - 1.3. **סוג ממשק לקליטה:** סוג הפלאגין אותו רוצים להפעיל.

מזהה חברה	סוג ממשק לקליטה	מזהה ספק	חברה	מס
1	SignatureR	623569617361764353	חברה לדוגמא	1
1	heshin	616231817616719873	חברה לדוגמא	2
1	reports	616231817616719873	חברה לדוגמא	3
1	kupain	616231817616719873	חברה לדוגמא	4
				5

תיקיית פלאג אין :

התיקייה נמצאת תחת: C:\hash7\H-Mobile\H-Mobile-1.0.8\Hconnect את שם קובץ ההמרה (Plugin) הרלוונטי יש להגדיר בסוג ממשק קולט.

2. צור קובץ אצווה "במוסופון 2" במבנה הבא:

batch file name: compld_interface_netPassportI

קובץ מיובא : c:\hash7\Hconnect\compld_interface_netPassportId_*.dat

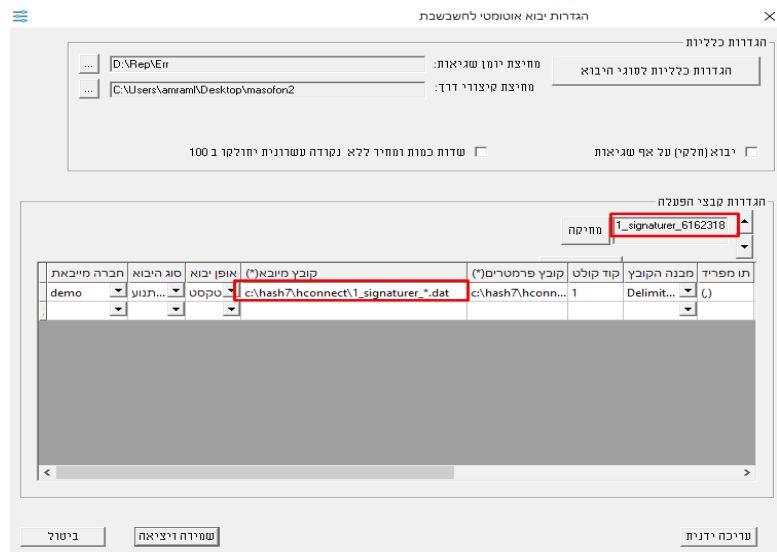
מבנה הקובץ: Delimited

תו מפריד: (,)

קובץ פרמטרים: *.prm

המרת עברית: Ansi2Ansi

לדוגמה:



API Post

Send post API request to <https://ws.wizground.com/api> In the following structure:

JS example:

```
fetch("https://ws.wizground.com/api",{
  headers: {
    "Content-Type":"application/json"
  },
  method: 'POST',
  body: JSON.stringify({
    "station": "4cf0a29b-5e20-4855-a46b-8293f31c1d59", // wanted station`s ID
    "plugin": "heshin", // wanted module`s name
    "company": "demo", // wanted company
    "message": { // message body:
      "netPassportID": "5884063480516097", // net-passport ID
      "pluginData": [{ // module data – json , array of objects
        "phone":"10",
        "stockid":"1111",
        "date":"1/298",
        "kabSort":"11",
        "docNumber":"5",
        "accountKey":444},
        {
          "phone":"10",
          "stockid":"1111",
          "date":"1/298",
          "kabSort":"11",
          "docNumber":"5",
```

```

"accountKey":444}}},
// signed data (explanation below)
"signature": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1ZXRQYXNzcG9yZdEiOiNTg4NDA2MzQ2ODgwNTE2MDk3IiwiaXNjaWVsbG8iLCJpYXQiOiJlY2MDA5NTY2MjJ9.Ez74J6dZjP-
chNoTZVRO24_e5wvSXxmX3FFB99vRY7F7AQQt_YcYdHcZj-9InfN7w7Tq-18zBGNDrvtnpBIXIiFmXkr7Q66VCas-gq9S-
alm4Z3jBFHPFFMFY3bfm_EDKKeqBVxC3o4pYdfJpIfYQNJ-6Yn-
lfw7fvDmHqnDDr0nmPwGhTvJCqu16ogLC24d0MVMINFFqDkD3sT-h6iWBcBunSd-
ZkczZpP0UVa6PXJoH494LPI8tn4vZruYLL0jhMNpEMB2pT48ugiljnK-GpzTRA67ZsJ73gT- -
HE6uSFVJMZ8Hd7FjEkv8PsH0ZWutj8WDEIG8DzHoZc28qJqBWPZsU43blmQyVXBrVOxZ7Xw6c0PeUzMcQkiHoxnD3c55m
LNxvNDuKuJBI0BBJjgy2iDdQf-L1denrqaacTtgPaFquNB-vtkxtYILStElc_2-50z- MKOjHvRtsWx2f_2nNivB0EXT-
A_kDJqn9r-AFXiJN3hgE_ISQ"
})
.then(response => response.json())
.then(result => console.log(result))
.catch(error => console.log(`{error} ${ error }`));

```

Sign data

There are two way to sign data , use netPassport in node.js or by EXE file

Node.js

- download your private key from netpassport.io .
- \$ npm install net-passport

```

const { signer } = require("net-passport");
const fs = require("fs");
// Pass in the .pem file or a path to the file
const pk = fs.readFileSync(
  path.join(__dirname, "lib", "keys", "privatekey.pem"),
  "utf-8"
);
// OR
const pk = path.join(__dirname, "lib", "keys", "privatekey.pem")
// Initiate your message object
const message = {
  netPassportID: 112233,
  myData: "Hi there"
};
// Pass in two parameters that includes your object message (as mentioned above) and
a private key or path to your private key

```

```
const signature = signer.sign(message, pk);
```

send the signature to the API.

Learn more about net-passport in <https://www.npmjs.com/package/net-passport>.

By exe file

download your private key from netpassport.io , and EXE file from:
for windows: <https://sws.wizground.com/files/netPassport-win.exe>
for mac: <https://sws.wizground.com/files/netPassport-macos>
for linux: <https://sws.wizground.com/files/netPassport-linux>

```
// send port as parameter to netPasport EXE and execute it
let port = 8040;
const exec = require("child_process").exec;
function createServer() {
  return new Promise((resolve, reject) => {
    var child = exec(`netPassport-win.exe ${port}`);

    child.stdout.on("data", data => {
      console.log(data);
      resolve("ok");
    });

    child.on("error", err => {
      console.log(`exe failed! , ${err}`);
      reject("err");
    });
  });
}

// send post message to http://localhost:yourPort/sign with the
// following parametrs for signing your data:
// privatekey - .pem file or a pth to the file.
// netPassportID - your net passport ID (from
// https://netpassport.io/home# on the top right).
// data - your data.
let signature = await axios.post(`http://localhost:${port}/sign`, {
  privatekey: "./privateKey.pem",
  netPassportID: "588406346880516097",
  data: JSON.stringify(modulData),
});

// to check if the signature is correct send post message to
// http://localhost:yourPort/verify with the following parametrs:
```

```
// privatekey - .pem file or a path to the file.  
// netPassportID - your net passport ID (from  
// https://netpassport.io/home# on the top right).  
// data - your data.  
//signature - the signature you got in sing request  
  
let verify = await axios.post(`http://localhost:${port}/verify`, {  
  privatekey: "./privateKey.pem",  
  netPassportID: "588406346880516097",  
  data: JSON.stringify(modulData),  
  signature: signature.data,  
});  
console.log(verify.data);
```

ERRORS

When something went wrong you will get response with status 500 .

information about the error is in response.apiRes.err .

API GET

לצורך משיכת נתונים מהחשבשבת המקומית ניתן ליצור דוח במחולל הדוחות "ולקרוא" לדוח דרך מערכת ה API

על מנת לקרוא לדו"ח במערכת ה API, יש להפעיל את ה command prompt

בתקייה C:\hash7\H-Mobile\H-Mobile-1.0.8\Hconnect

1. להפעיל את התוכנית Hrep
2. להגדיר את ה-ID של החברה ממנה רוצים למשוך את הנתונים (קוד חברה). לעזרה בנושא לחץ [באן](#)
3. להגדיר את ה-ID של הדוח שממנו רוצים לשלוף את הנתונים. לעזרה בנושא לחץ [באן](#)

```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.
C:\hash7\H-Mobile2\H-Mobile-1.0.8\Hconnect>hrep 1 185
```

לאחר מכן, יש להעתיק את השורות המצפנות של השאילתה שנוצרה, ולהדביק אותה בקוד ה API

encrypt_reportData

```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.
C:\hash7\H-Mobile2\H-Mobile-1.0.8\Hconnect>hrep 1 185

{"encrypt_reportData": "cf6c41d6ad9e3ee6429db2849738cd6:de39affd7118e52ab3e528a419b6658b9a8bca242add4f231437d3062bb5f36c
b792254dd1d4853a663f5cd899ab85ee6059508d2bfa959777b559b31a96cb2c909617079561cd34a47eccd9e0de79ae28c9c15a08e484be85523a94
6d0c758866627931185733c837eb603416f226b902d53cf7024b92dccc5565b38ee2ac15efe7c35ed414f2c1b246edfcfb0a59c9fd864e1e6a7c61
fa728f897692ae41043069352a20f2dbed6682e999b84e3591eed9f82ad2372d71db802cdec238d25ff5a0050fba3ae7d8ddfe7e0093b8f2bf314ebb
32d629263a15dd6a2daf46f4ed526f478b9415dd01e51d4fe98926f07b75af22b64425b2a6c689a022d3a3470a43c0eafae276f65b2fcb8b8c3239e
6a2a9fcbea6aefc460358345b1b2ae9caa12121cbe5a190fc7bb61065e262005ef63629164a0c41308daa977abf0d9cd5a22767d67eef16571ca2a46
22423d15654d3af14b1edc1a3b1574c78f69793adfee727f4906aa9c986d63abc9dbc01e7ce4e9ca5edc920ebcdc074cdc174323a667e8847cb7fd2
ae5ebc33a14e732f8389d02f1992ddd16722becc5973abb51b2f0a4958d84afacd4ffc3ef7b3bc229ed1285b0a57b358b9727409e11dbd2b43c6663
05b635ce1007e64c9f6422c624fcb4e7306ee22154c80e0f196e30c93decf6adfd7c30aafdd4a11ca16227416f607f2d47250a5d5eae232a7d11107
37e3aff49f67a6f32d4fa4ca2212a3ed09400f99b3faa523c0bec39504c944b89968be29781206ed7ff8102f47a194d997380db804b7b07535a1087d
c0270eb79f944c6ceda39879c97e7a68acdb6e2770b3e058cbbcbcab091233c21c8c92136b95443c3caa6c0fc823a419c56558d288f051c7fe8732ef8
bf90dc0ad6ff2fb5033cf96f3c80ef63c1b3437d25939be6373aa783e972cbbf5ced84350ef6c46ac5f67387bf62566a21de2785b9888fdde3f24ce
467b3c32374fab64867c05979bd2b1ac35c496f3ef81f4adf02033b06c31143009b35a51eb9040a4507c5d1c1ea6a88937c37e2879f8f53d7e6bfad7
0eae08893e7af5b241742cf1bbbedf1db358d9d000b1190f645ebbe1a58f609d", "params_data": []}
C:\hash7\H-Mobile2\H-Mobile-1.0.8\Hconnect>
```

```
const { signer } = require("net-passport");
const fs = require("fs");
const axios = require("axios");
var exec = require("child_process").exec;

// let modulData = [{ accountKey: "12345", docNumber: 80, transType: 33, suF: 34.54, creditFirm: "ויחה" }];
// modulData = [{ accountKey: "12345", exFile: "\\file\\1.png", EXPIC: "\\file\\2.pdf", VatExempt: 0 }];

const modulData = { "encrypt_reportData": "5260c06a6ade889dc227cbef7708d82e:62aa9393ad3dd7cc2f0985145b1e411c62b2016d9c5218d01e24f656f86c0" };

let port = 8040;

function createServer() {
  return new Promise((resolve, reject) => {
    var child = exec(`netPassport-win.exe ${port}`);

    child.stdout.on("data", data => {
      console.log(data);
      resolve("ok");
    });

    child.on("error", err => {
      console.log(`\n\t\tERROR: spawn failed! (${err})`);
      reject("err");
    });
  });
}
```

Reports JSON :

field	remarks	required	
encrypt_reportData		✓	מפתח לקוח
params_data		✓	מספר קבלה

1. The data in this file include two parts:

- **"encrypt_reportData"** – a long encrypted string that refers to the report structure.
- **"params_data"** - if the report was designed to be issued according to parameters, this part lists the parameters that you have to enter for a proper export of the data.

2. If you do use parameters you should update one property value only on each property object. The property "defVal" includes a default value for this property and this is the place to change it to the value that you want. All other properties should be modified according to their type. For example for report (below) having sort codes 100 - 399 you should change to "100" instead "999999999" and 399 instead "999999999".

- Date type parameter is in mm/dd/yy format.

Jason Example with "client" changes:

```
encrypt_reportData =
'Oc61e292d66a71326e011b089534ee1.....4b4312a84bac14e0ab520a0361d9bcd090';

params_data=[
  { "p_name": "__MUSTACH_PO__",
    "id": "0",
    "type": "long",
    "name": "חתך/שדה מרשומת החשבון/1. פרטי חשבון/קוד מיון",
    "defVal": "100",
    "opName": "מ.עד",
    "opOrigin": "from"
  },
  { "p_name": "__MUSTACH_P1__",
    "id": "500",
    "type": "long",
    "name": "חתך/שדה מרשומת החשבון/1. פרטי חשבון/קוד מיון",
    "defVal": "399",
    "opName": "מ.עד",
    "opOrigin": "to"
  }
];
```


Hconnect פלאגין הוספת

מחיצת הקבצים של Hconnect נמצאת תחת:

/hash7/H-Mobile/H-Mobile-currentVersion/Hconnect

כדי להוסיף פלאגין ל Hconnect יש צורך ל:

1. ליצור קובץ JSON עם הגדרה של הפרמטרים לממשק תחת

/hash7/H-Mobile/H-Mobile-currentVersion/Hconnect

לדוג' bankin.json

במבנה הבא:

```

{
  "data": [
    { "field": "Reference", "required": true, "type": "numeric", "length": 9 },
    { "field": "DatF", "required": true, "type": "date", "length": 10 },
    { "field": "Details", "required": false, "type": "alphaNumeric", "length": 50 },
    { "field": "SufDebit", "required": true, "type": "numeric", "length": 9.2 },
    { "field": "SufCredit", "required": true, "type": "numeric", "length": 9.2 },
    { "field": "Ballance", "required": false, "type": "numeric", "length": 9.2 },
    { "field": "SecDate", "required": false, "type": "date", "length": 10 },
    { "field": "PageNo", "required": false, "type": "numeric", "length": 3 }
  ]
}

```

Field types:

numeric – number (positive ,negative ,decimal).

alpha numeric – string.

boolean – true \ false.

date – date.

decimal – decimal , it's length refers to digits num before dot and after.

byte – bigger than (-128) and smaller than (128).

long - bigger than (-2,147,483,647) and smaller than (2,147,483,647).

int – Integer.

2. להוסיף בקובץ masofon.json את שם הממשק לדוג':

```
{ "hashModules": ["kupain", "invoice", "accounts", "heshin", "bankin" ] }
```

הקובץ נמצא ב:

/hash7/H-Mobile/H-Mobile-currentVersion/Hconnect

3. ביצירת קובץ BAT במסופון להגדיר :

שם הקובץ: compId_interface_netPassportId

קובץ מיובא: c:\hash7\Hconnect\compId_interface_netPassportId_*.dat

מבנה הקובץ: Delimited

תו מפריד: (,)

המרת עברית: Ansi2Ansi

הגדרת קובץ פרמטרים

הגדרת שמות הממשק בקבצים הנה זהה בכל המקומות

הגדרות כלליות

מחיצת יומן שגיאות: D:\Rep\Err

מחיצת קיצורי דרך: C:\Users\lanrami\Desktop\masofon2

יבוא (חלקי) על אף שגיאות

שדות כמות ומחיר ללא נקודה עשרונית יחולקו ב 100

הגדרות קבצי הפעלה

מחיקה

חברה מייבאת	סוג היבוא	אופן יבוא	קובץ מיובא (*)	קובץ פרמטרים (*)	קוד קולט	מבנה הקובץ	תו מפריד
demo	תנוע...	טקסט	c:\hash7\hconnect\1_signaturer_*.dat	c:\hash7\hconn...	1	Delimit...	(,)

ביטול שמירה ויציאה עריכה ידנית