

Vzdálená správa Procesu GROWE

Projekt GROWE 2012-18

VZDÁLENÁ SPRÁVA PROCESU GROWE

TÉMA: ŘÍZENÍ, VZDÁLENÁ SPRÁVA

VYTVOŘENO: 17.03.2018 19:08

VERZE: 1.0.3.

DATUM: 6.8.2023 22:14

EXPORT DO: -

PUBLIKOVÁNO: -

OBSAH

Vzdálená správa.....	3
Princip vzdálené správy – služba Server	3
Program Client Communicator.....	7
Konfigurace programu.....	8
Příkazy Client & Server	8
Popis komunikačních příkazů	8
Příkazy nevyžadující přihlášení (nekomunikující se službou Server).....	8
Příkazy nevyžadující přihlášení	9
Příkazy vyžadující přihlášení (chráněné).....	9
Uživatelské příkazy Serveru – podrobný popis.....	10
Příkaz Ping - #000.....	10
Příkaz Server Date - #001.....	10
Příkaz Server Time - #002	10
Příkaz Server Synchronize - #003.....	11
Příkaz Server Version - #004	11
Příkaz Server Info - #005.....	11
Příkaz Server Getlogfile - #006	11
Příkaz Server Getactiveacript - #007	11
Příkaz Server Changescript - #008	12
Příkaz Server Restartprocess - #009	12
Příkaz Server Down - #010.....	12
Příkaz Server Keylock - #011	12
Příkaz Server Keyunlock - #012	13
Příkaz Server Getcommandpack - #013	13
Příkaz Server Auto On - #014	13
Příkaz Server Auto Off - #015.....	13
Příkaz Server Gettemp - #016	14
Příkaz Server Control X Y - #017	14
Příkaz Server Testboard - #018	14
Systémové příkazy Serveru	14
Příkaz Server Text - #900	14
Příkaz - #950 (Test Spojeni).....	15
Příkaz Server Logout - #998	15
Příkaz Server Login - #999.....	15
LICENČNÍ ujednání	17

Licenční ujednání.....	17
Na základě této licenční smlouvy se udělují následující práva	17
Popis dalších práv a omezení.....	17
Záruky a odpovědnosti.....	17

VZDÁLENÁ SPRÁVA

Hlavní řídicí program **Procesu GROWE** je v podstatě bezobslužný, běžící zcela autonomně. Možnosti zásahu do samostatného procesu jsou omezeny, protože pro změnu jakýchkoli parametrů, nebo řídicího skriptu je potřeba řídicí program restartovat. Protože je praxí ověřeno, že takovéto zásahy jsou čas od času potřeba, byla pro tento účel vyvinuta **Vzdálená správa Procesu GROWE**.

PRINCIP VZDÁLENÉ SPRÁVY – SLUŽBA SERVER

Součástí řídicího programu je služba **Server** zajišťující komunikaci pomocí sítě **LAN**. Řídicí **PC** je nakonfigurováno tak, že v **OS** je vytvořen malý **RAM** disk o velikosti několika **MB**, sloužící jednak jako komunikační uzel (**Vertex**), jednak jako **File Handle Storage**, sloužící jako úložiště **HTML** stránek s informacemi z procesu **Řízení**.

Řídicí program při svém startu otestuje přítomnost tohoto **RAM** disku a vytvoří v něm **ServerRun** soubor, označující běh služby **Server** a její stav **On-Line**. Tento soubor je po celou dobu běhu služby uzamčen proti zápisu a smazání.

Pro připojení k **Serveru** slouží program **Client Communicator**, který je součástí instalačního balíku **Projekt GROWE** a jeho umístění je v adresáři ...\\GROWE\\RIZENI\\UTIL\\CLIENT.EXE.

```
Client Communicator, ver. 0.8.2. RC-I. build #235-01012018 build #235-01012018
Soucast projektu GROWE(c) 2012-18

#12.02.2017 - pridani prikazu SERVER PROGRAMS UPDATE
#06.12.2016 - pridana podpora barevneho zobrazeni vzdalene obrazovky
#27.10.2016 - pridani prikazu EXEC
#06.08.2016 - zmena Login procesu na jedno-pruchodovy
#18.07.2016 - pridani prepincu pri spusteni /UER
#29.06.2016 - opravena bezpecnostni chyba pri auto. provadeni skriptu
#12.05.2016 - pridani hlidac konektivity, prikaz Guard On/Off
#22.04.2016 - pridani podpory automatickych skriptu .SCR
#28.02.2016 - uprava navstevnického účtu GUEST
#09.11.2015 - update modulu .INIConfig I/O (uvodni INIRead), podpora zapisu
#31.10.2015 - zmena významu prikazu RESTARTPROCESS
#07.10.2015 - zmena prikazu SERVER - REBOOTMACHINE na DOWN
#15.09.2015 - rozsireni konfiguracnich hodnot nacitanych z .INI souboru
- pridani prikazu PREFIX - zmena obsahu rychle volby TAB
- zmena prikazu UER na VERSION a rozsireni vypisu o pouzite moduly

Detekce systemu ...
```

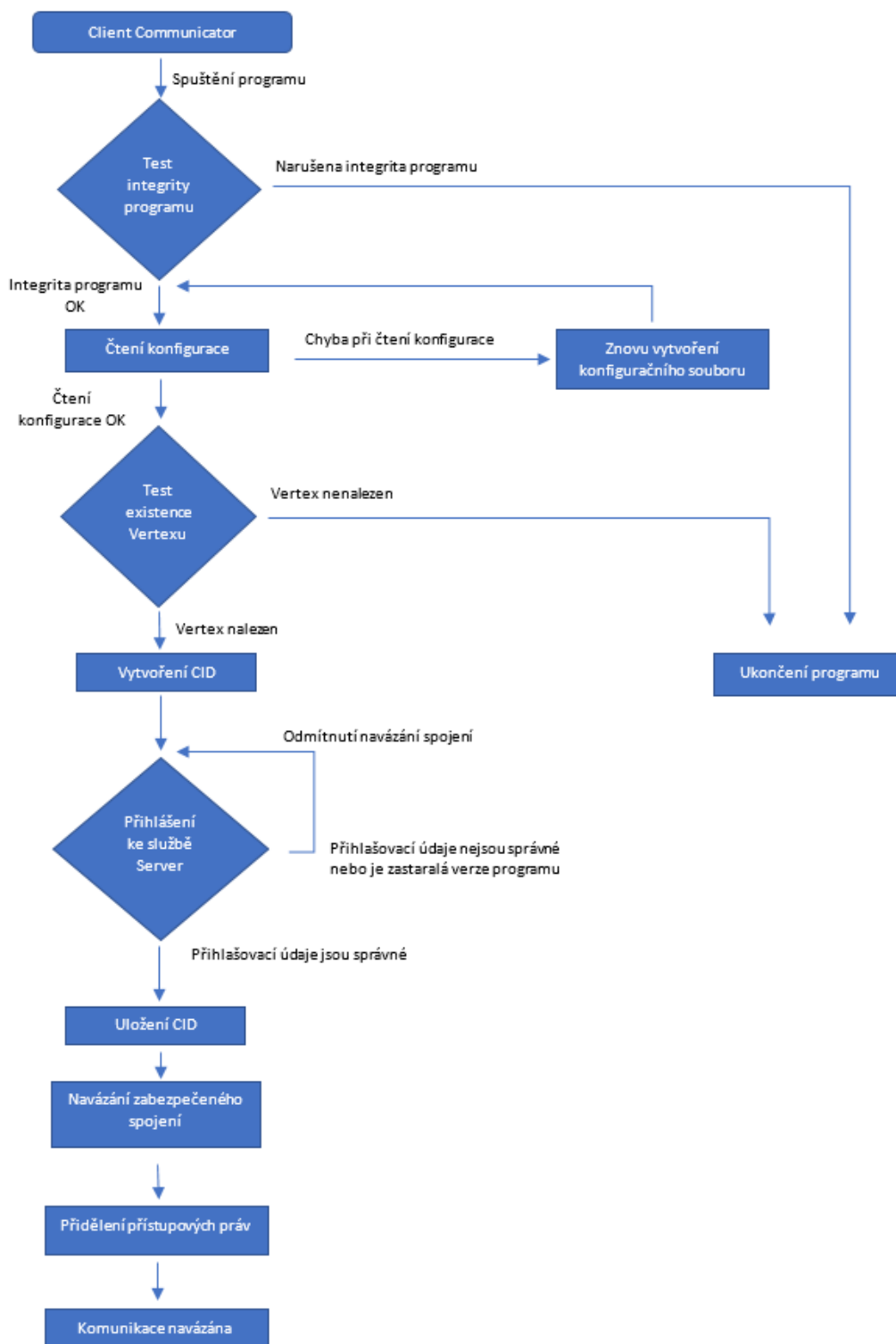
Spuštění programu Client Communicator

Komunikace mezi klientem a serverem probíhá pomocí komunikačního protokolu, který byl navržen právě pro tento účel. Pro komunikaci se používají komunikační pakety se šifrovaným obsahem. Ochrana navázané komunikace probíhá na několika úrovních.

Pro navázání komunikace je potřeba se přihlásit k serveru pomocí uživatelského jména a hesla. Anonymní připojení (bez přihlašovacích údajů) není možné. **Client Communicator** při svém startu vytvoří jedinečný bezpečnostní klíč **CID (ClientID)**, který je platný po celou dobu spuštění programu. Při každém spuštění se generuje klíč nový. Tento klíč se předá straně **Server** společně s přihlašovacími údaji. **Server** údaje zkontroluje a pokud odpovídají formátu komunikačního protokolu a jsou správné, vytvoří spojení. Po vytvoření spojení se dle přístupových práv uživatele zpřístupní sada příkazů, kterými lze jednak zjišťovat stav **Procesu GROWE** a jednak jej ovlivňovat, nebo měnit jeho konfiguraci. Součástí každého paketu s navázaným spojením je i klíč **CID**, který zabezpečuje, že do komunikace nevstoupí neoprávněně třetí strana. Pokud **Server** nalezne neplatný, nebo chybějící klíč **CID**, komunikace se z bezpečnostních důvodů přeruší.

Další ochranou je zabezpečení proti **Hammer Attack**, kde se cyklicky testují různé přihlašovací údaje (např. pomocí nejčastěji používaných už. jmen a hesel. Pokud služba **Server** zjistí vícenásobný neúspěšný pokus o přihlášení, deaktivuje se a dále je vzdálená správa nedostupná. Znovu spuštění služby je pak potřeba provést přímo na řídicím **PC**, kdy je uživatel (na obrazovce **Debug**) informován o této události.

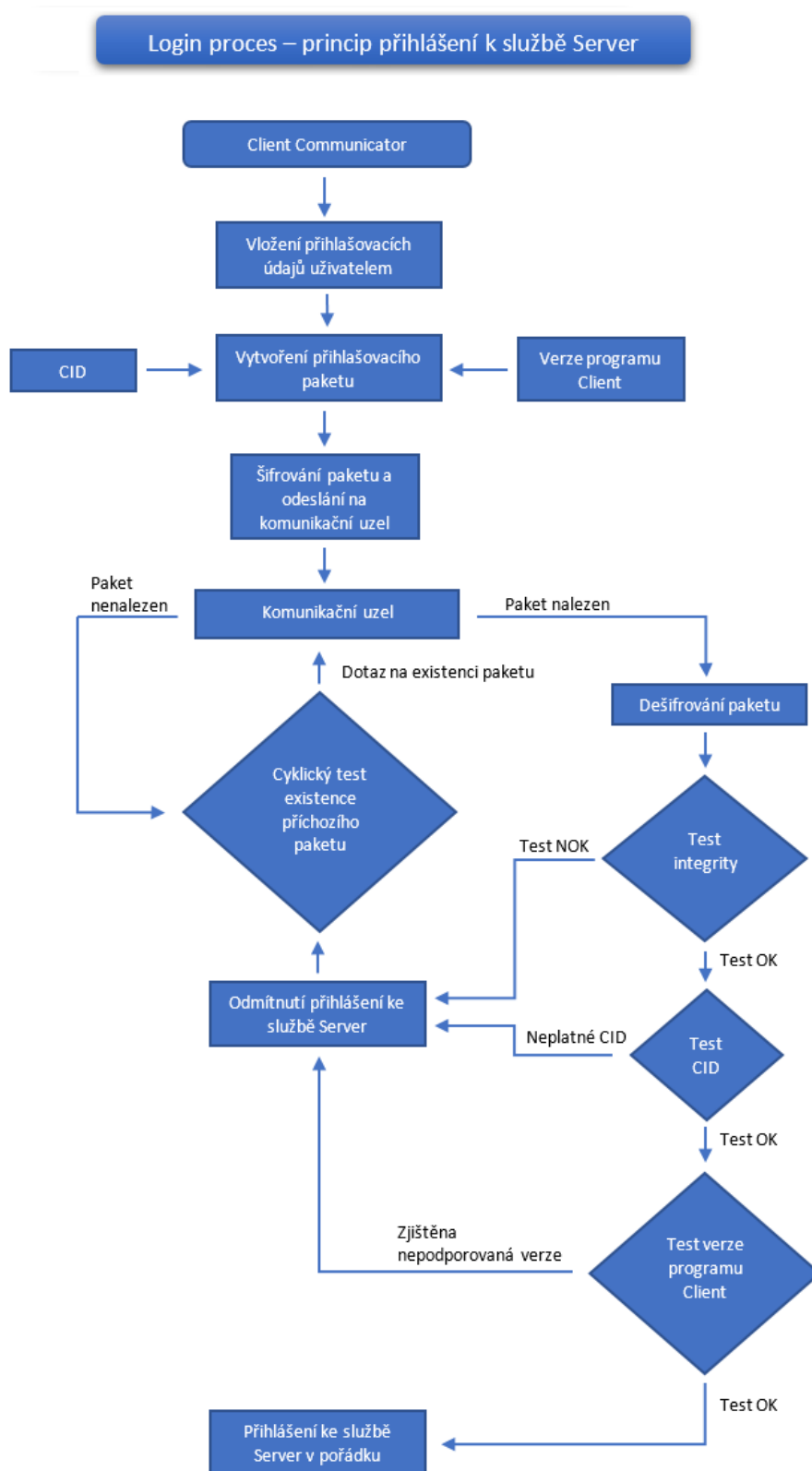
Řídící proces – princip navázání spojení



Mapa procesů při navazování zabezpečené komunikace

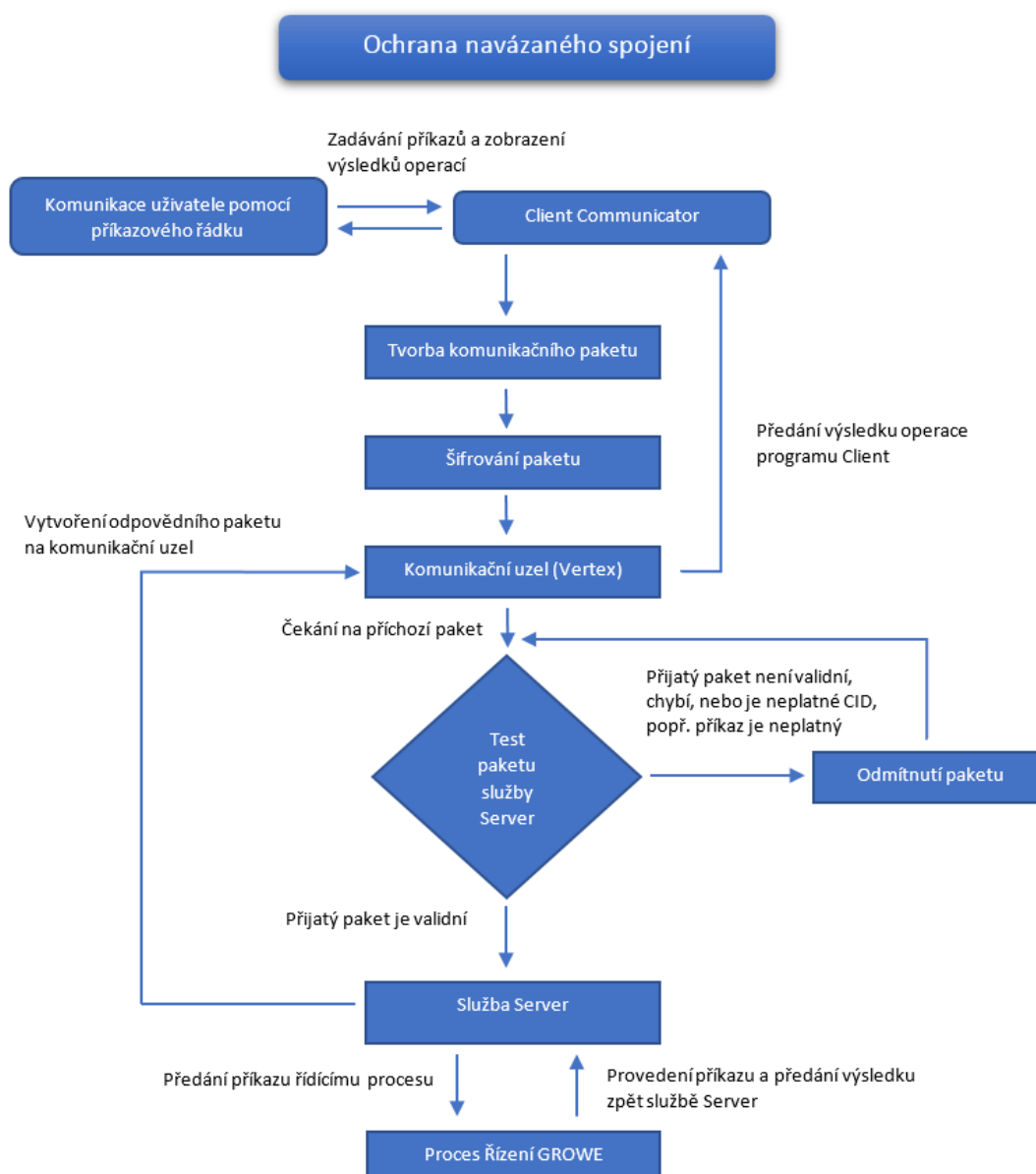
Proti vzájemné nekompatibilitě stran **Server** a **Client** odesílá při přihlašovacím procesu číslo své verze. Pokud je verze klienta nižší, než je minimální požadovaná, k navázání spojení nedojde jeho odmítnutím na straně serveru. Tato ochrana je implementována z důvodu rozšiřování sady příkazů a zajištění, že obě strany používají stejnou verzi komunikačního protokolu. Jeho současná nejaktuálnější je verze **SRV_Cmmnd_v0.99_E**.

Princip navazování zabezpečené komunikace je znázorněna na mapě procesů. Každý z klíčových kroků je kontrolován na bezchybnost, pokud se vyskytne chyba při navazování spojení, je proces přihlašování přerušena a služba **Server** je uvedena do výchozího stavu.



Mapa procesů – princip ověřování přihlášení

Samotná komunikace je pak zabezpečena proti nežádoucímu průniku nebo narušení, jak je zobrazeno v mapě procesů zabezpečení spojení.



Mapa procesu zabezpečení již navázaného spojení

PROGRAM CLIENT COMMUNICATOR

Program **Client Communicator** slouží ke vzdálené komunikaci se službou **SERVER Communicator**. Tato služba je svázaná programově s programem pro řízení procesu GROWE a umožňuje jeho vzdálenou správu a monitoring. Pro komunikaci je využita síť **MS-NET** protokol **TCP/IP** s využitím technologie tzv. **Comm. Vertex** – komunikačního uzlu.

```
Client Communicator ver. 0.8.2. RC-I. build #235-01012018
Soucast projektu GROWE(c) 2012-18

pro dalkovou spravu procesu GROWE(c)
pomoci technologie Client & Server Communicator
```

Client Communicator – úvodní hlášení

Komunikační uzel je vytvořen jako **RAM** disk sdílený na straně **SERVERu**, o velikosti cca. **4 MB**. Na tento uzel se zasílají komunikační pakety, v nichž jsou předávány příkazy nebo žádosti o službu na straně **SERVER**. Komunikační paket je podrobně popsán v souboru **PACKET.TXT**, princip komunikace je v souboru **COMMUNIC.TXT**. Po odeslání příkazového paketu 'naslouchá' klient na komunikačním uzlu, dokud **SERVER** neodešle odpověď. Poté tuto zobrazí uživateli, popř. ji zpracuje do výstupních dat, které poté zobrazí.

Existuje několik úrovní komunikace. Nechráněná a chráněná, a tato se dále liší podle přístupových práv. Nechráněná komunikace umožňuje pouze zjištění existence komunikačního uzlu, stavu serveru **ON Line/OFF**

```
Client Communicator, ver. 0.7.1b, build #142
Soucast projektu GROWE(c) 2012-15

#15.09.2015 - pridan prikaz PREFIX - zmena obsahu rychle volby TAB.
              - zmena prikazu VER na VERSION a rozsireni vypisu o pouzite moduly.
#07.10.2015 - rozsireni konfiguracnich hodnot nacistanych z .INI souboru.
#31.10.2015 - zmena vyznamu prikazu RESTARTPROCESS.
              - zmena prikazu SERVER - REBOOTMACHINE na DOWN.
#09.11.2015 - update modulu .INIConfig I/O (puvodni INIRead), podpora zapisu.

Detekce systemu ...
Detekovano DOSBox virtualni prostredi.
Komunikace v soucasne dobe neni funkcní v tomto prostredi z duvodu chyby
opetovneho vytvoreni cache disku (refresh obsahu) az po navratu do SHELL

Program ukoncen.

Windows neni spusten.
Detekovana sit ... neni detekovana.
Vychazi pismeno komunikacniho uzlu je nastaveno na M:\

Pro napovedu pouzij prikaz HELP.
[Anonymous]>
```

Client Communicator (50-ti řádkový mód)

Line, rychlost odezvy serveru a přihlášení pomocí uživatelského účtu. Teprve úspěšné přihlášení umožňuje využívat pokročilé příkazy, sloužící jednak ke správě služby **Server** a dále skrze tuto službu ovládat další, v tomto případě službu **ŘÍZENÍ**. Komunikace probíhá v krytované podobě, a to zašifrováním těla paketu. Po přihlášení k službě **Server** navíc dochází ke kontrole tzv. **CID – ClientID**. **ClientID** je řetězec 15 náhodně vygenerovaných znaků, který se vytvoří po startu programu a je platný po celou dobu jeho běhu. Po ukončení a opětovném spuštění se vytvoří nové **CID**. Toto **CID** slouží k ochraně komunikace po přihlášení, aby nemohlo dojít k převzetí vyšších oprávnění. Client odesílá **CID** v každém komunikačním paketu, **Server** kontroluje jeho shodu s **CID** použitým při přihlašování. Pokud **SERVER** nalezne neshodu, přeruší navázané spojení a provede vnitřní znovunastavení do výchozího stavu. Každá taková zachycená neshoda se považuje za závažné narušení ochrany komunikace, pokud se objeví opakovaně, předpokládá se, že probíhá pokus o převzetí nebo neoprávněného

zásahu do komunikace třetí stranou a služba se trvale deaktivuje. Obdobně **Server** kontroluje vícenásobné neúspěšné přihlášení, kdy toto považuje za tzv. **HAMMER ATTACK** (násilné prolomení založené na opakovaném zkoušení přihlašovacích údajů).

KONFIGURACE PROGRAMU

Pro správnou funkci vzdálené správy je potřeba nakonfigurovat komunikační program i uzel. Konfigurační soubor programu **Client Communicator** používá formát **.INI**.

```
[INI]
[Configuration]
colorfg=7
colorbg=1
vertexletter=m
prefixtab=server
clientlines=50
waitstateforserver=180
```

Popis jednotlivých konfiguračních proměnných hlavního programu:

colorfg – Barva pozadí, hodnota v rozsahu **0** až **15**

colorbg – Barva popředí, hodnota v rozsahu **0** až **15**

vertexletter – písmeno komunikačního uzlu

prefixtab – často se opakující příkaz, slovo, které se vloží do řádku po stisku klávesy **TAB**

clientlines – počet řádků, které zobrazuje program **Client Communicator**, hodnota je **25** nebo **50**

waitstateforserver – časová prodleva, po kterou program čeká na odpověď serveru, výchozí je **180**

PŘÍKAZY CLIENT & SERVER

Příkazová sada, která se používá pro obousměrnou komunikaci je dle specifikace **SERVER COMMAND**.

Informace o specifikaci:

Verze příkazové sady **SERVER COMMAND**, ver. **0.9.9. BETA**

SYSTEM Client & Server GROWESYSTEM Communication

Značka příkazové sady: **SRV_Cmmnd_v0.99_E**

POPIS KOMUNIKAČNÍCH PŘÍKAZŮ

Tento soubor slouží zároveň jako návodný text, kdy při požadavku na server o zaslání informací o specifikaci příkazové sady, se uloží na komunikační uzel tento soubor.

Všechny příkazy jsou odesílány klientem na komunikační uzel serveru, kde jsou dále zpracovány. Příkazy jsou rozděleny do dvou sad – veřejné a chráněné, přístupné až po úspěšném přihlášení k službě server. Veřejné příkazy slouží k základnímu zjištění stavu služby serveru (stav **ON LINE/OFF LINE**, test rychlosti odezvy, zvýšení uživatelských práv) a stavu komunikačního uzlu. Chráněné příkazy jsou dostupné až po úspěšném přihlášení k serveru, kdy celá přihlašovací procedura je na straně serveru a klient pouze přijímá textové výzvy k vložení už. jména a hesla a předává tyto získané údaje od uživatele. Pokud není uživatel přihlášen, je provedení všech chráněných příkazů odmítnuto serverem.

PŘÍKAZY NEVYŽADUJÍCÍ PŘIHLÁŠENÍ (NEKOMUNIKUJÍCÍ SE SLUŽBOU SERVER)

- COMM** - dotaz na konektivitu serveru
NETVER - informace o detekované verzi sítě LAN

PŘÍKAZY NEVYŽADUJÍCÍ PŘIHLÁŠENÍ

- PING** - ověření komunikace se serverem
SERVER LOGIN - přihlášení k serveru (získání vyšších oprávnění pro použití chráněných příkazů)

PŘÍKAZY VYŽADUJÍCÍ PŘIHLÁŠENÍ (CHRÁNĚNÉ)

Server Date	- vrací systémové datum serveru
Server Time	- vrací systémový čas serveru
Server Text	- pošle zprávu na obrazovku serveru
Server Synchronize	- synchronizuje datum a čas serveru podle PC klienta
Server Version	- vrací informace o verzi serveru
Server Info	- vrací podrobné informace o serveru
Server GetLOGFiles	- uloží .LOG soubory serveru na Vertex
Server GetActiveScript	- uloží aktivní .DBF skript na Vertex
Server ChangeScript	- vymění aktivní skript za nový (dotaz na soubor)
Server RestartProcess	- restartuje proces GROWE
Server Down	- deaktivuje službu SERVER
Server KeyLock	- vypíná službu pro čtení klávesnice serveru
Server UnLock	- zapíná službu pro čtení klávesnice serveru
Server GetCommandPack	- odešle popis příkazové sady serveru na vertex
Server LogOut	- odhlášení od serveru (odebrání vyšších oprávnění)
Server Auto On	- zapnutí auto řízení procesu GROWE
Server Auto Off	- vypnutí auto řízení procesu GROWE
Server GetTemp	- vrátí poslední čtenou teplotu procesu GROWE
Server Control X Y	- pošle číslo X na Relé Y řídící desky
Server TestBoard	- auto test řídící reléové desky

UŽIVATELSKÉ PŘÍKAZY (čísla 0-899)

písemná forma	číselná forma	datum vytvoření	platnost
Ping	#000	* 28. 12. 2014	PLATNÝ
Server Date	#001	* 01. 2015	PLATNÝ
Server Time	#002	* 01. 2015	PLATNÝ
Server Synchronize	#003	* 01. 2015	PLATNÝ
Server Version	#004	* 01. 2015	PLATNÝ
Server Info	#005	* 01. 2015	PLATNÝ
Server GetLogFile	#006	* 01. 2015	PLATNÝ
Server GetActiveScript	#007	* 01. 2015	PLATNÝ
Server ChangeScript	#008	* 01. 2015	PLATNÝ
Server RestartProcess	#009	* 20. 06. 2015	PLATNÝ
Server RestartMachine	#010	* 01. 2015	DEAKTIVOVÁN
Server Down	#010	* 31. 10. 2015	PLATNÝ
Server KeyLock	#011	* 01. 2015	PLATNÝ
Server KeyUnLock	#012	* 01. 2015	PLATNÝ
Server GetCommandPack	#013	* 05. 2015	PLATNÝ
Server Auto ON	#014	* 10. 07. 2015	PLATNÝ
Server Auto OFF	#015	* 10. 07. 2015	PLATNÝ
Server GetTemp	#016	* 10. 07. 2015	PLATNÝ
Server Control X Y	#017	* 10. 07. 2015	PLATNÝ
Server TestBoard	#018	* 11. 07. 2015	PLATNÝ

SYSTÉMOVÉ PŘÍKAZY (rezervovaná část, čísla 900-999)

písemná forma	číselná forma	datum vytvoření	platnost
Text	#900	* 04.2015	PLATNÝ
GetScreen	#910	* 11. 07. 2015	PLATNÝ
[test spojení]	#950	* 26. 06. 2015	PLATNÝ
Debug On/Off	#980	* 11. 07. 2015	DEAKTIVOVÁN
LogOut	#998	* 01. 02. 2015	PLATNÝ
LogIn	#999	* 01. 02. 2015	PLATNÝ

UŽIVATELSKÉ PŘÍKAZY SERVERU – PODROBNÝ POPIS

Všechny argumenty uzavřené do závorek [] jsou nepovinné, argument Vertex musí být zapsán ve formátu písmena RAM disku + ":" -> M:

PŘÍKAZ PING - #000

Ověření komunikace se serverem

Syntaxe příkazu:

PING vertex: [/C]

[/C] - cyklická operace, přerušeno po ESC

(stejný příkaz se používá se pro udržování navázaného spojení se serverem)

NOPP vertex

Definice komunikace:

Klient odešle příkaz "000"

Server přijme příkazový paket, odešle odpověď "Communication OK".

Klient přijme paket odpovědi a zobrazí text.

Přepínač [/C] způsobí cyklické volání, přerušeno kl. ESC.

Příkaz NOPP cyklické volání neumožňuje.

PŘÍKAZ SERVER DATE - #001

Vrací systémové datum PC serveru, vyžaduje přihlášení

Syntaxe příkazu:

Server DATE

Definice komunikace:

Klient odešle příkaz "001".

Server přijme příkaz, odešle systémové datum PC ve formátu "mm-dd-rrrr".

Klient přijme paket odpovědi a zobrazí text odpovědi.

PŘÍKAZ SERVER TIME - #002

Vrací systémový čas PC serveru, vyžaduje přihlášení

Syntaxe příkazu:

Server TIME

Definice komunikace:

Klient odešle příkaz "002".

Server přijme příkaz, odešle systémový čas PC ve 24 hod. formátu "hh-mm-ss".

Klient přijme paket odpovědi a zobrazí text odpovědi.

PŘÍKAZ SERVER SYNCHRONIZE - #003

Synchronizuje systémový datum a čas serveru podle času klienta

Syntaxe příkazu:

Server SYNCHRONIZE

Definice komunikace:

Klient odešle příkaz "003" + systémový datum PC ve formátu "mm-dd-rrrr" + systémový čas PC ve 24 hod. formátu "hh-mm-ss".

Server přijme příkaz, aktualizuje sys. datum a čas dle přichozích informací a odešle textovou zprávu "Sync OK".

Klient přijme paket odpovědi a zobrazí text odpovědi.

PŘÍKAZ SERVER VERSION - #004

Vrací informaci o verzi serveru

Syntaxe příkazu:

Server VERSION

Definice komunikace:

Klient odešle příkaz "004".

Server přijme příkaz, odešle textovou zprávu obsahující informaci o verzi služby server.

Klient přijme paket odpovědi a zobrazí text odpovědi.

PŘÍKAZ SERVER INFO - #005

Server uloží podrobné informace o svém běhu na vertex v .LOG souboru

Syntaxe příkazu:

Server INFO

Definice komunikace:

Klient odešle příkaz "005".

Server přijme příkaz, zkopíruje soubor SERVER.LOG z lokálního adresáře na vertex a odešle textovou zprávu "Save File To Comm Vertex"

Klient přijme paket odpovědi a na dotaz zobrazí soubor odpovědi.

Klient na dotaz smaže .LOG soubor odpovědi z vertexu. Při opakovaném příkazu se .LOG soubor na vertexu přepisuje.

PŘÍKAZ SERVER GETLOGFILE - #006

Server uloží na vertex všechny nalezené .LOG soubory z lokálního adresáře

Syntaxe příkazu:

Server GETLOGFILES

Definice komunikace:

Klient odešle příkaz "006".

Server přijme příkaz, zkopíruje soubory *.LOG z lokálního adresáře na vertex a odešle textovou zprávu "Send xLOGFilesOK", kdy x označuje počet odeslaných souborů.LOG

Klient přijme paket odpovědi a zobrazí text odpovědi.

Soubory .LOG jsou stále uloženy na vertexu, pokud je příkaz volán opakovaně jsou soubory přepisovány (aktualizovány).

PŘÍKAZ SERVER GETACTIVEACRIPT - #007

Uloží aktivní řídicí .DBF skript na vertex

Syntaxe příkazu:

Server GETACTIVESCRIPT

Definice komunikace:

Klient odešle příkaz "007".

Server přijme příkaz, nalezne (podle informací z řídicího systému) aktivní řídicí skript .DBF z lokálního adresáře, tento zkopíruje na vertex a odešle textovou zprávu "SendActivScr".

Poté server odešle textovou zprávu o výsledku operace restartu procesu.

Soubor skriptu .DBF je stále uložen na veretxu, pokud je příkaz volán opakovaně, je soubor přepisován.

Pokud server nenalezne soubor skriptu .DBF v aktivním adresáři odešle textovou zprávu "Not Available Script".

PŘÍKAZ SERVER CHANGESCRIPT - #008

Zaměnění aktivní řídicí .DBF skript na serveru

Syntaxe příkazu:

Server CHANGESCRIPT

Definice komunikace:

Klient zobrazí dotaz na název souboru, kterým nahradí původní skript na serveru.

Po vybrání souboru je provedena jeho kontrola, zda je skript validní (kontrola platnosti digitálního podpisu). Pokud není kontrola ukončena s chybou, klient zkopíruje skript na vertex, odešle příkaz "008"+ název skriptu, server přijme paket, přesune skript z vertexu do lokálního adresáře a předá pokyn řídicímu programu pro záměnu skriptu v procesu ŘÍZENÍ.

Poté server odešle textovou zprávu o výsledku operace záměny skriptu a restartuje se.

PŘÍKAZ SERVER RESTARTPROCESS - #009

Restartuje řídicí proces GROWE

Syntaxe příkazu:

Server RESTARTPROCESS

Definice komunikace:

Klient odešle příkaz "009"

Server přijme příkaz, předá požadavek o restart procesu ŘÍZENÍ.

Poté server odešle textovou zprávu o výsledku operace restartu procesu "System SERVER now restarting..."

SERVER provede interní restart, odpojí připojenou relaci a uvede se do výchozího stavu

Klient přijme paket s odpovědí, zobrazí text odpovědi a provede automatické odhlášení

PŘÍKAZ SERVER DOWN - #010

Deaktivuje službu SERVER (při běhu procesu ŘÍZENÍ)

Syntaxe příkazu:

Server DOWN

Definice komunikace:

Klient odešle příkaz "010"

Server přijme příkaz, předá požadavek o deaktivaci služby nadřazenému programu

Poté server odešle textovou zprávu o přijetí požadavku o deaktivaci služby.

Klient přijme paket odpovědi a zobrazí text odpovědi.

PŘÍKAZ SERVER KEYLOCK - #011

Zamyká klávesnici serveru (SERVER odpojí službu čtení klávesnice)

Syntaxe příkazu:

Server KEYLOCK

Definice komunikace:

Klient odešle příkaz "**011**".

Server přijme příkaz, provede zamčení klávesnice a vrátí textovou zprávu "**Server Keyboard Lock**".

Klient přijme paket s odpovědí a zobrazí text odpovědi.

PŘÍKAZ SERVER KEYUNLOCK - #012

Odemyká klávesnici serveru (SERVER připojí službu čtení klávesnice)

Syntaxe příkazu:

Server KEYUNLOCK

Definice komunikace:

Klient odešle příkaz "**012**".

Server přijme příkaz, provede odemčení klávesnice a vrátí textovou zprávu "**Server Keyboard Unlock**".

Klient přijme paket odpovědi a zobrazí text odpovědi.

PŘÍKAZ SERVER GETCOMMANDPACK - #013

Server vyhledá v domácím adresáři soubor SRVCMNT.TXT a uloží ho na vertex

Syntaxe příkazu:

Server GETCOMMANDPACK

Definice komunikace:

Klient odešle příkaz "**013**".

Server přijme příkaz, provede kontrolu existence souboru, pokud existuje, zkopíruje ho na vertex a vrátí textovou zprávu o úspěšném dokončení operace "**Server Send Command Pack to Vertex**", jinak odešle textovou zprávu o neúspěchu "**Command Pack not Found**".

PŘÍKAZ SERVER AUTO ON - #014

Příkaz předá procesu ŘÍZENÍ požadavek o zapnutí automatické regulace dle řídicího skriptu

Syntaxe příkazu:

Server AUTO ON

Definice komunikace:

Klient odešle příkaz "**014**".

Server přijme příkaz, zapne službu automatického řízení procesu GROWE. Vrátí textovou zprávu o úspěšném dokončení operace "**Server Auto On Ready**", jinak odešle textovou zprávu o neúspěchu "**Server Auto ON Fail**".

PŘÍKAZ SERVER AUTO OFF - #015

Příkaz předá procesu ŘÍZENÍ požadavek o vypnutí automatické regulace dle řídicího skriptu

Syntaxe příkazu:

Server AUTO OFF

Definice komunikace:

Klient odešle příkaz "**015**".

Server přijme příkaz, vypne službu automatického řízení procesu GROWE. Vrátí textovou zprávu o úspěšném dokončení operace "**Server Automatic Control is OFF**", jinak odešle textovou zprávu o neúspěchu "**Server Auto OFF Fail**".

PŘÍKAZ SERVER GETTEMP - #016

Vrátí hodnotu poslední měřené teploty v procesu GROWE

Syntaxe příkazu:

Server GETTEMP

Definice komunikace:

Klient odešle příkaz "**016**".

Server přijme příkaz a vrátí textovou zprávu s informací o aktuální platné teplotě. Jinak odešle textovou zprávu o chybě při čtení teploty.

PŘÍKAZ SERVER CONTROL X Y - #017

Příkaz předá procesu ŘÍZENÍ požadavek na reset, změnu stavu nebo vrátí aktuální stav všech relé.

Syntaxe příkazu:

Server CONTROL X Y

Definice komunikace:

Klient odešle příkaz "**017**", dále hodnoty **X** a **Y**.

Server přijme příkaz, předá procesu GROWE požadavek o změnu na relé **X** stav **Y**. Rozsah hodnoty relé **X** je **0–9**, kdy **0** provede reset všech relé do stavu **VYPNUTO**, hodnota **9** vrátí pouze stav všech relé, bez toho, že by byly provedeny změny. V obou těchto případech je druhá hodnota **Y** ignorována. V případě použití hodnoty **1** až **8** tím se označí, na kterém relé má být provedena změna. V tomto případě druhá hodnota označuje, stav, na který má být současný změněn. Hodnota **0** znamená vypnuto, **1** zapnuto a **2** změna současného stavu na opačný. Jiné hodnoty jsou ignorovány.

UPOZORNĚNÍ:

*Před tímto příkazem je nutno vypnout automatické řízení, příkazem **SERVER AUTO_OFF**, protože každá další událost dle řídicího skriptu může změnit požadovanou hodnotu na původní v něm uvedenou. K této kontrole stavu dochází 1x za sec. a může dojít ke krátkodobému vypnutí relé, kdy se přeruší napájení, což může narušit funkcionality některých spotřebičů v **Procesu GROWE**.*

PŘÍKAZ SERVER TESTBOARD - #018

Příkaz předá procesu ŘÍZENÍ požadavek na auto test řídicí reléové desky, vrátí výsledek testu.

Syntaxe příkazu:

Server TESTBOARD

Definice komunikace:

Klient odešle příkaz "**018**".

Server přijme příkaz, předá procesu GROWE požadavek provedení auto testu. Při jeho průběhu jsou postupně spínána jednotlivá relé a kontroluje se vždy jejich stav. Poté je provedena kontrola bloků současně sepnutých relé 4 a 5. Po ukončení testu SERVER vrátí informaci o výsledku testu.

SYSTÉMOVÉ PŘÍKAZY SERVERU

PŘÍKAZ SERVER TEXT - #900

Odeslání textového řetězce a jeho zobrazení na obrazovce serveru

Syntaxe příkazu:

SERVER TEXT

Definice komunikace:

Klient odešle příkaz **"900"** + **text zprávy**

Server přijme příkaz, zobrazí text zprávy na obrazovku serveru a vrátí textovou zprávu **"Incoming message akcepted."** v paketu odpovědi.

Klient přijme paket a zobrazí text odpovědi.

PŘÍKAZ - #950 (TEST SPOJENÍ)

Automaticky příkaz, nelze vyvolat pomocí příkazu v SHELLu klienta, spouští automaticky klient po úspěšném přihlášení k serveru, v cyklu á 10 sec.

Syntaxe příkazu:

Klient odešle příkaz **"950"** (automatický příkaz systému)

Definice komunikace:

Pro tento speciální příkaz je zkrácena doba čekání na odpověď a to na 1/3 této doby, pokud server neodpoví, předpokládá se, že spojení bylo ztraceno a klient se uvede do stavu nepřipojeno a změní **PROMPT** na **<[Anonymous]>**

PŘÍKAZ SERVER LOGOUT - #998

odhlášení od serveru (ukončení konekce, ztráta vyšších oprávnění)

Syntaxe příkazu:

Server LOGOUT

Definice komunikace:

Klient odešle příkaz **"998"**.

Server přijme příkaz, provede odhlášení klienta od serveru a vrátí textovou zprávu **"User XXXXX disconnect from server."**, kdy **XXXXX** je uživatelské jméno odhlašovaného uživatele.

Klient přijme paket a zobrazí text odpovědi a změní **PROMPT** na **<[Anonymous]>**

PŘÍKAZ SERVER LOGIN - #999

Přihlášení uživatele k serveru (navázání konekce, získání vyšších oprávnění)

Syntaxe příkazu:

Server LOGIN

Definice komunikace:

Klient zobrazí dotaz na název písmene vertexu, dojde o ověření platnosti vertexu, klient odešle příkaz **"999"** + **CCVer**(Verze klienta) + **CID**(ClientID)

Server přijme příkaz, uloží **CCVer** a **CID** do paměti a nastaví příznak pro probíhající proces **LOGIN**. Server odešle dotaz na uživatelské jméno formou textového dotazu **"User name:"** a **"Password:"** v paketu. Zadávatel hesla probíhá v režimu skrytých znaků na obrazovce, poté vše odešle v **Login** paketu serveru. Server přijme paket, uloží do paměti, zkontroluje platnost podpory klienta **CCVer**, uloží klíč **CID**, dekryptuje **LOGIN Skript** – soubor **PASSWORD.TXT**, zkontroluje platnost už. jména a hesla. Při všech shodách nastaví příznak pro přihlášení, odešle text odpovědi **"User XXXXX logged sucessfully."**, kdy **XXXXX** je uživatelské jméno přihlašovaného uživatele. Dále zpřístupní chráněné příkazy serveru. Při použití těchto příkazů jsou dále kontrolovány informace **CCVer** a **CID** aby nemohlo dojít k převzetí navázané komunikace s jiným klientem nebo cizím programem a s ní neoprávněnému navýšení práv k serveru. Po úspěšném přihlášení změní klient **PROMPT** na **<[Logged]>**. Pokud se přihlašovací údaje (už. jméno nebo heslo) neshodují, server odmítne konekci a odešle odpověď **"Login fail."**

Pokud se neshoduje **CCVer**, server odešle odpověď **"Wrong Client or unsupported Client version. Login fail."**

Pokud se neshoduje **CID**, což je závažná bezpečnostní chyba, která může nastat např. po přihlášení klientem k serveru, poté restartu klienta a pokusu užití statusu přihlášení (neoprávněné navýšení práv k serveru). Server uloží do **RUN.LOG** souboru hlášení o pokusu o narušení bezpečnosti "**Pokus o neautorizovaný přístup k serveru**" a ukončí konekci a odpojí službu **SERVER**. Tato služba se pak musí ručně restartovat na řídícím **PC**. Na stranu klienta se neodesílají o tomto žádná chybová hlášení, aby nedocházelo k nechtěnému napomáhání při pokusu o násilné prolomení prozrazením reakcí systému. Klient zachytí přerušení konekce a změní **PROMPT** na **<[Anonymous]>**.

LICENČNÍ UJEDNÁNÍ

Autorská práva, pokud nejsou výslovně uvedena jinak, náleží společnosti **Projekt GROWE**. Autorská práva náležející k programovým modulům, které jsou použity, jsou ve zdrojových kódech uvedeny, včetně jejich vlastníků.

LICENČNÍ UJEDNÁNÍ

Toto licenční ujednání je právní smlouvou mezi Vámi, koncovým spotřebitelem (dále jen uživatel) softwarového produktu **Client Communicator**, (dále označován jako **produkt**) a **společností Projekt GROWE®** (v textu použito jako **Projekt GROWE**). Produkt jako takový se neprodává, pouze se uděluje licence k jeho užívání. Instalací a používáním produktu uživatel potvrzuje, že souhlasí s podmínkami tohoto ujednání a zavazuje se jimi řídit.

NA ZÁKLADĚ TÉTO LICENČNÍ SMLOUVY SE UDĚLUJÍ NÁSLEDUJÍCÍ PRÁVA

Produkt je poskytnut uživateli **ZDARMA**, počet instalací, nebo kopií není nijak omezen. Může jej bez omezení šířit, pokud dodrží pravidlo, že **program bude šířen jako celek, včetně podpůrných, konfiguračních a dalších souborů, úplné a nepozměněné dokumentace a elektronických manuálů**.

POPIS DALŠÍCH PRÁV A OMEZENÍ

Uživatel není oprávněn produkt prodávat, pronajímat, půjčovat nebo poskytovat produkt za úplatu. Zpětná analýza produktu není povolena. Jednotlivé komponenty nebo části produktu nelze oddělovat pro samostatné použití. Je zakázáno, upravovat chod programu prostřednictvím produktů třetích stran. Zasahovat do databáze (import/export nebo zpracovávání dat) je povoleno, avšak s přihlédnutím na možného zásah do autorských práv jejich vlastníků.

Projekt GROWE si vyhrazuje právo na změnu licenční politiky bez předchozího upozornění.

ZÁRUKY A ODPOVĚDNOSTI

Produkt poskytován bez jakýchkoli záruk. Projekt GROWE pak zvláště nezaručuje bezvadný chod produktu, pokud:

- a) je provozován spolu s programy jiných výrobců, které znemožňují bezchybný chod programů
- b) je provozován na chybně nakonfigurovaném počítači nebo na chybně nastavené počítačové síti
- c) uživatel provedl zásah do produktu, do jeho databázových nebo konfiguračních souborů pomocí jiných nástrojů, nebo prostředků, než je samotný program nebo nástroje, které jsou součástí programového vybavení a pro tento zásah určeny.

Projekt GROWE nenese odpovědnost za případné následné škody, ať jsou jakékoli, které by vznikly na základě použití jeho produktů. **Projekt GROWE** nenese odpovědnost za použití softwarového vybavení k jiným účelům, než je zde výslovně uvedeno. V případě požadavku na satisfakci za možné vyniklé škody, náleží uživateli částka do maximální výše nákladů, prokazatelně uhrazených za softwarový produkt k rukám společnosti **Projekt GROWE**.