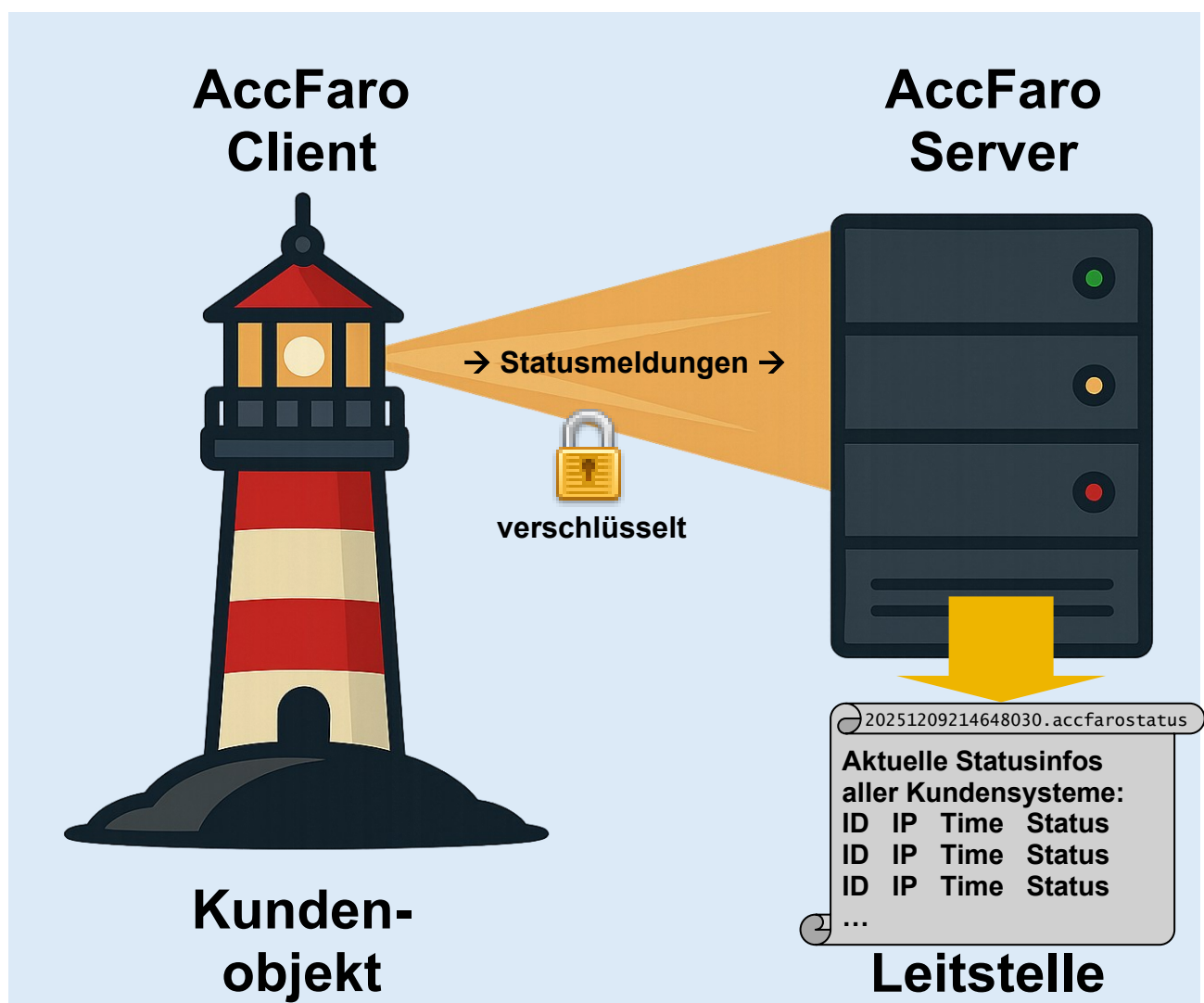


AccFaro

IP-Adressen und Status entfernter Systeme ermitteln



Status: Entwurf

Dieses Dokument ist geistiges Eigentum der Accellence Technologies GmbH und darf nur mit unserer ausdrücklichen Zustimmung verwendet, vervielfältigt oder weitergegeben werden

Accellence Technologies GmbH • Garbsener Landstr. 10 • 30419 Hannover
Tel. 0511 277-2400 • Fax 0511 277-2499 • E-Mail: info@accellence.de • Website: www.accellence.de

Inhalt

1	Einleitung.....	3
1.1	Ausgangslage und Aufgabe	3
1.2	Die Lösung: Leuchtturm und Leitstelle	4
1.3	Architektur und Erweiterbarkeit	5
1.4	Syntaxhinweise	5
2	Der AccFaroServer in der Leitstelle.....	5
2.1	Server starten.....	6
2.2	Clients anlegen und Schlüssel verwalten	7
2.3	Alternative: Client mit Einmalcode anmelden	8
2.4	Zustand der Clients überwachen.....	9
2.5	Empfangene Meldungen anzeigen.....	10
2.6	Konfiguration des Servers	11
2.7	Erreichbarkeit: Port und Firewall	13
2.8	Als Windows-Dienst installieren	13
3	Der AccFaroClient auf den überwachten Systemen	14
3.1	Client-Paket importieren	15
3.2	Alternative: Anmelden mit Einmalcode.....	16
3.3	Status und Test	17
3.4	Inhalt der Meldung.....	18
3.5	Konfiguration des Clients	19
3.6	Als Windows-Dienst installieren	19
4	Verwaltung mit dem AccMicroServiceKonfigurator.....	20
5	Format der Statusdateien	21
6	Erkennen und Melden von Systemausfällen	22
6.1	Lebenszeichen für AccRoutineMonitor	22
7	Überwachung der Alarmwege	23
8	Datensicherheit	23
9	Was tun wenn	25
10	Redundanter Betrieb	27
11	Digitale Souveränität	27
12	Download und Updates	28
13	Support / Hotline.....	28

1 Einleitung

Mit **AccFaro** – bestehend aus den beiden Diensten **AccFaroClient** und **AccFaroServer** – erfährt eine Leitstelle jederzeit, **welche Systeme erreichbar sind, unter welcher IP-Adresse** sie gerade zu finden sind und **in welchem Zustand** sie sich befinden – auch wenn deren IP-Adressen nicht fest sind oder nicht per DNS ermittelt werden können.

Der AccFaroClient läuft auf jedem zu überwachenden System und sendet regelmäßig (z.B. alle 30 Sekunden) eine kurze, verschlüsselte Meldung an den AccFaroServer in der Leitstelle. Der Server merkt sich die jeweils letzte Meldung jedes Systems und schreibt daraus in festem Intervall eine Statusdatei, die andere Anwendungen auswerten können – z.B. um in EBÜS eintreffende Alarme dem richtigen System zuzuordnen oder um Ausfälle zu erkennen.

Beide Dienste gehören zu den **AccMicroServices** und werden wie diese mit dem **AccMicroServiceKonfigurator** verwaltet (siehe → www.ebues.de/AccMicroServices.pdf). Jeder Dienst hat eine eigene Weboberfläche (Web-GUI) für Status, Konfiguration und – beim Server – die Verwaltung der Clients.

Die Gliederung dieses Dokuments folgt dem empfohlenen Vorgehen: Zuerst den Server in der Leitstelle einrichten und die Clients anlegen (Kapitel 2), dann die Clients auf den überwachten Systemen in Betrieb nehmen (Kapitel 3).

1.1 Ausgangslage und Aufgabe

Leitstellen müssen viele Systeme überwachen, die sich oft an den unterschiedlichsten Standorten befinden. Dazu sind IP-Verbindungen zwischen der Leitstelle und all diesen Systemen erforderlich. Viele dieser Systeme haben keine feste IP-Adresse, bei manchen kann die Adresse nicht einmal per DNS ermittelt werden.

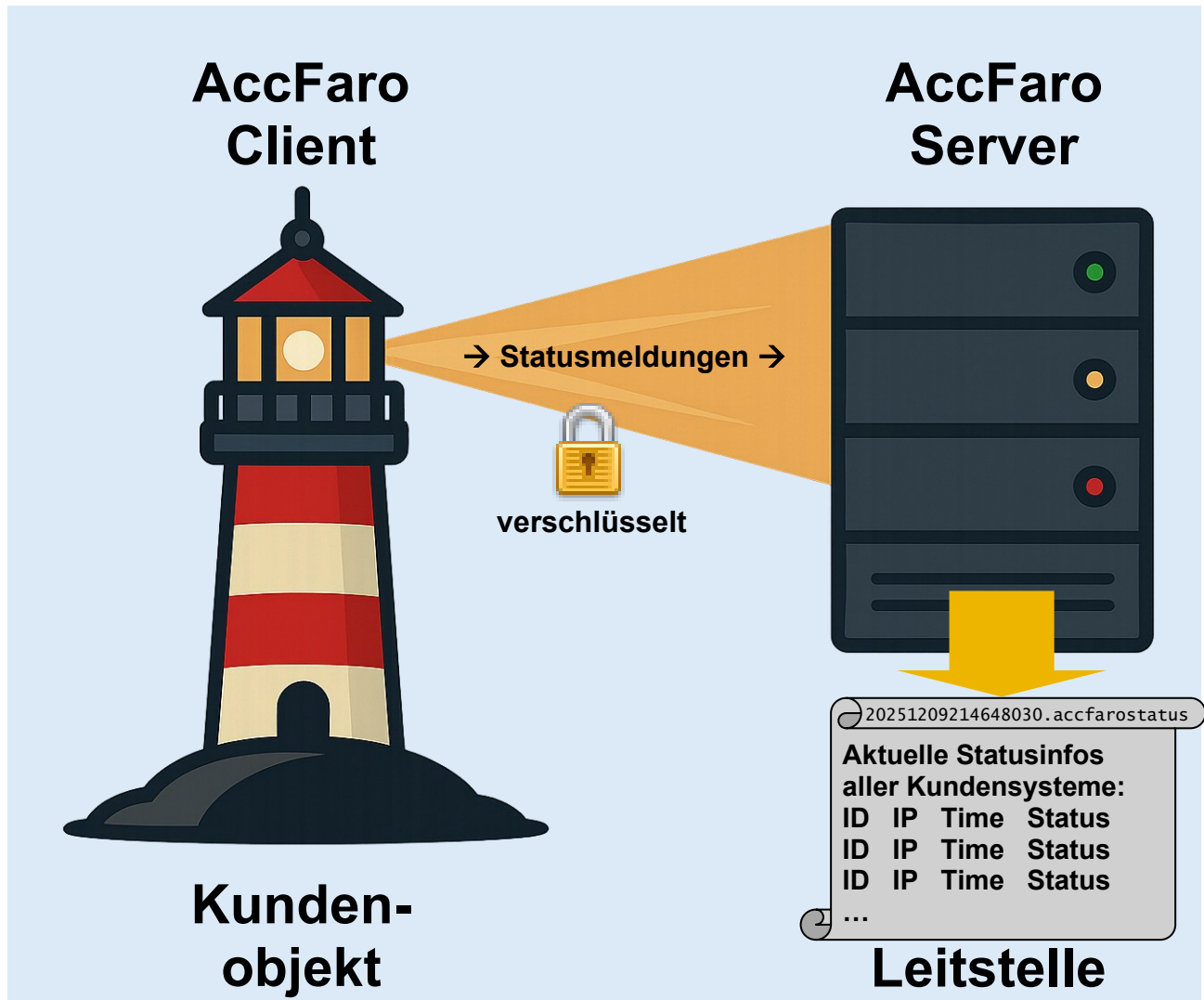
Damit Leitstellen die Systeme ihrer Kunden aufschalten können, müssen die IP-Adressen all dieser Systeme bekannt sein, um

- via IP in der Leitstelle eintreffende Alarme dem richtigen System zuzuordnen und
- von der Leitstelle aus Verbindungen zu diesen Systemen aufzubauen.

Gesucht war deshalb eine Lösung, mit der Leitstellen die IP-Adressen aller aufzuschaltenden Systeme ermitteln können, auch wenn diese nicht bekannt sind. Dabei sollen weitere benötigte Daten der Systeme mitgeliefert werden, und anhand der zyklisch gemeldeten Daten soll die Leitstelle die Betriebsbereitschaft der Systeme und die Netzwerkverbindung zu ihnen überwachen können.

1.2 Die Lösung: Leuchtturm und Leitstelle

Auf den zu überwachenden Systemen läuft ein einfacher, kompakter Dienst, der wie ein Leuchtturm (spanisch: Faro) regelmäßig eine kurze Statusmeldung an die Leitstelle sendet: der **AccFaroClient**.



Die Statusmeldung enthält u. a. Informationen, mit denen die Leitstelle das System identifizieren kann (ID). Die ID kann aus mehreren Parametern zusammengesetzt werden; in EBÜS sind das z. B. der Name des Schutzobjektes (*so*) und der Name der Bildquelle (*bq*). Beim Empfang einer Meldung ermittelt der Server die IP-Adresse des Absenders und ordnet sie über die ID genau diesem System zu.

In der Leitstelle läuft dazu der **AccFaroServer**. Er empfängt die Meldungen aller AccFaroClients und erstellt daraus eine minutenaktuelle Tabelle mit den IDs, IP-Adressen und – bei Bedarf – weiteren Daten sämtlicher Systeme. Weitere Software in der Leitstelle kann diese Tabelle nutzen, um eintreffende Alarmer korrekt zuzuordnen und externe Systeme gezielt aufzuschalten und zu überwachen.

1.3 Architektur und Erweiterbarkeit

Im Sinne einer schlanken, robusten und verteilten Microservice-Architektur kümmern sich die AccFaro-Dienste um genau diese Aufgaben und um nichts anderes. Bei Bedarf können auf beiden Seiten weitere Anwendungen oder Dienste ergänzt werden, die z. B.

- aktuelle Statusinformationen des überwachten Systems in die Meldungsdatei `message.inf` eintragen, damit sie mit den Meldungen zur Zentrale verschickt werden,
- Daten aus den ermittelten Tabellen automatisch in die Konfiguration der Bildquellen in der Leitstelle übernehmen, z. B. die aktuellen IP-Adressen,
- anhand der gemeldeten Daten die Funktion aller Systeme und Netzwerke überwachen und Störungen aktiv melden – und vieles mehr.

Die Verbindung zwischen Client und Server nutzt die gemeinsame, gehärtete Netzwerkschicht der AccMicroServices (Kapitel 8). Sie ist so ausgelegt, dass künftig auch dauerhafte Verbindungen für weitere Anwendungen darauf aufsetzen können.

Hinweis: Geprüfte Plattform ist Windows. Der Quellcode ist plattformneutral gehalten, damit Linux und – für den Client – später auch Android folgen können; diese Portierungen sind noch nicht ausgeliefert und nicht getestet.

1.4 Syntaxhinweise

<x> Platzhalter, für den konkrete Werte eingesetzt werden müssen.

» Verweis auf weitere Dokumente oder Textstellen.

Schaltfläche Bezeichnung einer Schaltfläche in einer Weboberfläche.

2 Der AccFaroServer in der Leitstelle

Nach der Installation von EBÜS finden Sie den AccFaroServer im Verzeichnis

```
C:\EBUES\AccMicroServices\AccFaroServer
```

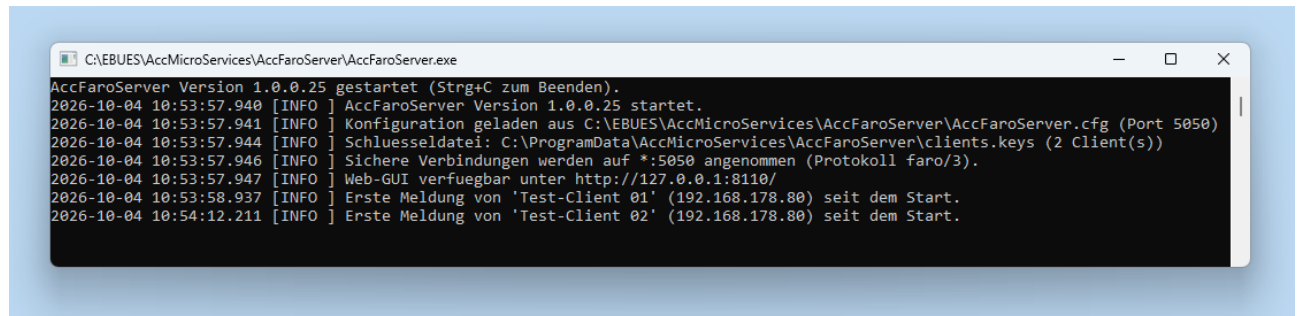
Der Ordner enthält neben der Anwendung (`AccFaroServer.exe`) zwei Textdateien:

- `AccFaroServer.cfg` – die Konfiguration (Port, Schutzgrenzen, Statusverzeichnis, ...)
- `message.inf` – die Vorlage der Quittung, die jeder Client nach seiner Meldung zurückbekommt

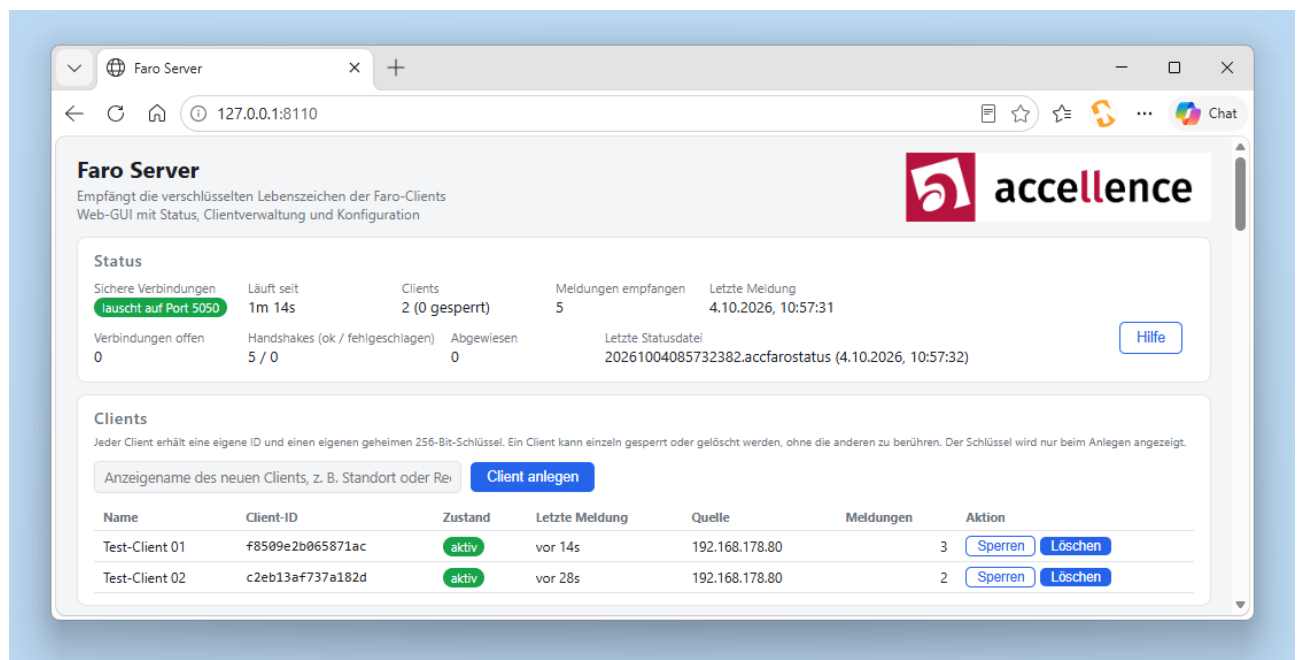
Beide Dateien lassen sich bequem über die Weboberfläche bearbeiten. Zusätzlich legt der Server – **außerhalb** des Programmordners – eine Schlüsseldatei an (Abschnitt 2.2).

2.1 Server starten

Führen Sie `AccFaroServer.exe` durch Doppelklick aus. Es öffnet sich ein Konsolenfenster, in dem alle Status- und Fehlermeldungen direkt angezeigt werden – besonders hilfreich für Test- und Diagnosezwecke. Ist in der Konfiguration `guiautoplay=1` gesetzt (werkseitig), öffnet sich zusätzlich automatisch die Weboberfläche des Servers im Browser (Standard: `http://127.0.0.1:8110/`).



Der Server wird mit **Strg+C** im Konsolenfenster wieder beendet. Für den Dauerbetrieb installieren Sie ihn als Windows-Dienst (Abschnitt 2.8).



Der Bereich **Status** zeigt auf einen Blick, ob der Server für sichere Verbindungen bereitsteht (Port), seit wann er läuft, wie viele Clients angelegt sind, wie viele Meldungen eingegangen sind und – für die Beurteilung der Schutzmechanismen – wie viele Verbindungen offen, fehlgeschlagen oder sofort abgewiesen wurden. Kann der Port nicht geöffnet werden (z.B. weil er belegt ist), steht dort die Ursache; der Server versucht es alle 10 Sekunden erneut.

2.2 Clients anlegen und Schlüssel verwalten

Jedes System, das Meldungen senden soll, ist im Server als **Client** angelegt und erhält dabei eine **eigene ID** und einen **eigenen geheimen Schlüssel** (256 Bit, zufällig erzeugt). Dadurch kann jeder Client einzeln gesperrt oder gelöscht werden, ohne dass andere Clients neu eingerichtet werden müssen.

1. Im Bereich **Clients** der Server-Weboberfläche einen **Anzeigenamen** eingeben (z. B. den Namen des Standorts oder des Rechners).
2. Auf **Client anlegen** klicken.
3. Es erscheint das **Client-Paket**: ein kurzer Text mit Serveradresse, Port, Client-ID und Schlüssel. Mit **In Zwischenablage kopieren** oder **Als Datei speichern** sichern Sie es für die Einrichtung des Clients (Abschnitt 3.1).

Faro Server
Empfängt die verschlüsselten Lebenszeichen der Faro-Clients
Web-GUI mit Status, Clientverwaltung und Konfiguration

Status

Sichere Verbindungen lauscht auf Port 5050	Läuft seit 2m 38s	Clients 3 (0 gesperrt)	Meldungen empfangen 11	Letzte Meldung 4.10.2026, 10:59:02
Verbindungen offen 0	Handshakes (ok / fehlgeschlagen) 11 / 0	Abgewiesen 0	Letzte Statusdatei 20261004085903412.accfarostatus (4.10.2026, 10:59:03)	

Clients
Jeder Client erhält eine eigene ID und einen eigenen geheimen 256-Bit-Schlüssel. Ein Client kann einzeln gesperrt oder gelöscht werden, ohne die anderen zu berühren. Der Schlüssel wird nur beim Anlegen angezeigt.

Anzeigenname des neuen Clients, z. B. Standort oder Rechner **Client anlegen**

Client-Paket – nur jetzt sichtbar
Dieses Paket enthält den geheimen Schlüssel. Es wird nur jetzt angezeigt und kann später nicht erneut abgerufen werden. Auf dem Client in der Web-GUI des AccFaroClients unter „Client-Paket importieren“ einfügen. Danach hier schließen; das Paket nicht per E-Mail oder Chat weitergeben.

```
AccFaroClient-Paket v1
server=LAPNAU07
serverport=5050
clientid=d84d1697699cb256
key=ihfd7u/fnd1sdbpKsgg6&dsfhdNQ=
```

In Zwischenablage kopieren **Als Datei speichern** **Schließen**

Name	Client-ID	Zustand	Letzte Meldung	Quelle	Meldungen	Aktion
Test-Client 01	f8509e2b065871ac	aktiv	vor 7s	192.168.178.80	6	Sperr Löschen
Test-Client 02	c2eb13af737a182d	aktiv	vor 22s	192.168.178.80	5	Sperr Löschen
Test-Client 03	d84d1697699cb256	noch keine Meldung	–	–	0	Sperr Löschen

Hinweis: Das Client-Paket enthält den **geheimen Schlüssel**. Es wird nur beim Anlegen angezeigt und kann später **nicht erneut abgerufen** werden – geht es verloren, legen Sie einen neuen Client an und löschen den alten. Geben Sie das Paket nicht unverschlüsselt per E-Mail oder Chat weiter.

Aus Sicherheitsgründen lassen sich Client-Schlüssel nur erzeugen, solange die Weboberfläche des Servers auf 127.0.0.1 beschränkt ist (Standard); andernfalls würde der Schlüssel unverschlüsselt per HTTP übertragen. Die Schlüssel selbst stehen **nicht** in der Konfigurationsdatei, sondern in einer Schlüsseldatei, auf die nur SYSTEM, die Administratoren und das Dienstkonto zugreifen können:

C:\ProgramData\AccMicroServices\AccFaroServer\clients.keys

(Bei einer benannten Instanz heißt der Ordner AccFaroServer.<Instanzname>.) Behandeln Sie diese Datei – und eine Sicherung davon – wie ein Passwort.

2.3 Alternative: Client mit Einmalcode anmelden

Lässt sich das Client-Paket nicht übertragen – zum Beispiel weil der Client in einer Kamera, einem Videorecorder oder einem Alarmsystem steckt, das keine E-Mails empfangen kann und keine USB-Sticks zulässt –, melden Sie den Client stattdessen mit einem **Einmalcode** an. Dafür genügt ein **Telefonat**: Es muss nichts übertragen oder kopiert werden. Das Client-Paket (Abschnitt 2.2) bleibt daneben unverändert möglich.

1. Im Bereich **Clients** der Server-Weboberfläche einen **Anzeigenamen** eingeben (leer lassen, dann übernimmt der Server den Rechnernamen des Geräts) und auf **Mit Einmalcode anlegen** klicken.
2. Es erscheint ein **16-stelliger Code** (z. B. 4821 0937 5516 2084) mit Server, Port und der Restzeit (10 Minuten). Sagen Sie dem Anwender **Code, Server und Port** am Telefon durch. Der Code gilt nur **einmal**.
3. Der Anwender trägt sie an seinem Client unter **Anmelden mit Einmalcode** ein (Abschnitt 3.2) und nennt Ihnen die dort angezeigte **Bestätigungszahl** (4 Ziffern).
4. Im Bereich **Anmeldungen mit Einmalcode** erscheint die wartende Anmeldung mit Quelle (IP), Rechnernamen und **Bestätigungszahl**; im Titel der Browser-Registerkarte steht die Zahl der wartenden Anmeldungen, z. B. „(1)“. Vergleichen Sie die Zahl, und klicken Sie nur dann auf **Bestätigen**, wenn sie übereinstimmt und Sie gerade mit dem Anwender sprechen. Im Zweifel: **Ablehnen**.
5. Erst jetzt erzeugt der Server den dauerhaften Schlüssel und überträgt ihn verschlüsselt an den Client. Der Client erscheint in der Liste der Clients und beginnt zu melden.

Der Bereich **Anmeldungen mit Einmalcode** zeigt zu jedem Code den Zustand: **wartet auf das Gerät** (Code ausgegeben, noch nicht benutzt – mit **Abbrechen** lässt er sich zurückziehen), **wartet auf Ihre Bestätigung, angemeldet, abgelehnt, abgelaufen** (der Code wurde nicht innerhalb der Frist benutzt bzw. nicht rechtzeitig bestätigt), **abgebrochen** oder **fehlgeschlagen** (z. B. wenn die Verbindung beim Übergeben abbricht; der Server legt dann keinen Client an). Beendete Einträge bleiben zwei Minuten sichtbar.

Hinweis: Der Code allein genügt nicht: **Ohne Ihre Bestätigung erhält niemand einen Schlüssel** – auch nicht, wer den Code mithört. Die Bestätigungszahl wird aus der verschlüsselten Verbindung abgeleitet; hat jemand den Code zuerst benutzt oder sich dazwischengeschaltet, stimmen die Zahlen nicht überein. Der Code besteht aus 16 Ziffern, davon eine Prüfziffer, die Tippfehler am Client sofort erkennt. Er gilt einmal und nur für `enrollvalidseconds` (Standard 600 Sekunden); wird er mehrfach falsch eingegeben (`enrollmaxfailures`, Standard 20), zieht der Server alle unbenutzten Codes zurück. Der Server erzeugt Codes nur, solange seine Weboberfläche auf 127.0.0.1 beschränkt ist; das Bestätigen ist auch über den AccWebUiGateway möglich.

2.4 Zustand der Clients überwachen

Die Tabelle im Bereich **Clients** zeigt für jeden Client:

Spalte	Bedeutung
Name / Client-ID	Anzeigenname und die zufällige, nicht aussagekräftige Kennung des Clients.
Zustand	aktiv – die letzte Meldung ist jünger als die eingestellte Frist. überfällig – seit mehr als staleafterseconds (Standard 120 s) keine Meldung. gesperrt – der Client wird abgewiesen. noch keine Meldung – angelegt, aber seit dem Start des Servers nichts empfangen.
Letzte Meldung	Alter der letzten Meldung. Führt man mit der Maus darüber, erscheint ihr Inhalt; übersichtlicher zeigt ihn der Bereich Meldungen (nächster Abschnitt).
Quelle	IP-Adresse, von der die letzte Meldung kam.
Meldungen	Anzahl der seit dem Start empfangenen Meldungen.

Clients

Jeder Client erhält eine eigene ID und einen eigenen geheimen 256-Bit-Schlüssel. Ein Client kann einzeln gesperrt oder gelöscht werden, ohne die anderen zu berühren. Der Schlüssel wird nur beim Anlegen angezeigt.

Anzeigenname des neuen Clients, z. B. Standort oder Re...

Client anlegen

Name	Client-ID	Zustand	Letzte Meldung	Quelle	Meldungen	Aktion
Test-Client 01	f8509e2b065871ac	gesperrt	vor 5m 29s	192.168.178.80	21	Entsperren Löschen
Test-Client 02	c2eb13af737a182d	aktiv	vor 13s	192.168.178.80	31	Sperren Löschen
Test-Client 03	d84d1697699cb256	noch keine Meldung	–	–	0	Sperren Löschen
Test-Client 04	7d4b917473a533ec	überfällig	vor 2m 8s	192.168.178.80	1	Sperren Löschen

Mit **Sperren** wird ein Client sofort abgewiesen (ohne Neustart); sein Schlüssel bleibt gespeichert, und **Entsperren** hebt die Sperre wieder auf. **Löschen** entfernt den Client samt Schlüssel endgültig. Für den Client sieht ein gesperrter oder gelöschter Zugang genauso aus wie ein falscher Schlüssel (Kapitel 8).

Der Zustand „überfällig“ ist im Server nur eine Anzeige. Eine automatische Alarmierung bei Ausfällen übernehmen weitere Anwendungen anhand der Statusdateien (Kapitel 6).

2.5 Empfangene Meldungen anzeigen

Der Bereich **Meldungen** zeigt die zuletzt empfangene Meldung jedes Clients übersichtlich als Tabelle – denselben Stand, den der Server in die Statusdatei schreibt (Kapitel 5). Die Felder der Meldung sind dabei in eigene Spalten zerlegt:

Spalte	Bedeutung
Client	Anzeigenname des Clients (bei einem gesperrten Client mit Kennzeichnung).
Empfangen / Alter	Zeitpunkt des Empfangs (Ortszeit des Browsers) und wie lange das her ist. Das Alter ist grün, solange die Meldung jünger als <code>stafterseconds</code> ist, danach gelb („überfällig“).
Quelle	IP-Adresse, von der die Meldung kam.
Host, Host-UUID, Host-Status	Die Felder <code>host_name</code> , <code>host_uuid</code> und <code>host_status</code> aus der Meldung (Abschnitt 3.4).
Zeit laut Client	Das Feld <code>timecode_utc17</code> der Meldung, in Ortszeit umgerechnet. Weicht es deutlich vom Empfangszeitpunkt ab, geht die Uhr des Clients falsch.
Quittung	Was der Server dem Client geantwortet hat: OK, oder bei einer fehlerhaften Meldung z. B. <code>NO_MESSAGE_UUID</code> (rot).
Anzahl	Anzahl der seit dem Start des Servers empfangenen Meldungen dieses Clients.

Die Tabelle **aktualisiert sich selbstständig** alle wenige Sekunden (abschaltbar mit **Automatisch aktualisieren**; **Aktualisieren** lädt sofort neu). Darüber steht, welche Statusdatei zuletzt geschrieben wurde (Name, Zeitpunkt, Anzahl der Zeilen) und wann die Tabelle zuletzt aktualisiert wurde. Ein Klick auf eine **Spaltenüberschrift** sortiert die Tabelle (erneuter Klick kehrt die Reihenfolge um); das Feld **Suchen** blendet alle Zeilen aus, in denen der Suchbegriff weder im Namen, in der IP, in den Host-Feldern, in der Quittung noch im Text der Meldung vorkommt. Mit ► wird der vollständige, unveränderte Text der Meldung unter der Zeile eingeblendet. Sortierung, Suche und aufgeklappte Meldungen bleiben bei der automatischen Aktualisierung erhalten.

Meldungen

Die zuletzt empfangene Meldung jedes Clients - derselbe Stand, den der Server in die Statusdatei schreibt. Die Tabelle aktualisiert sich selbstständig; ein Klick auf eine Spaltenüberschrift sortiert, ► zeigt den vollständigen Text der Meldung.

☒ Automatisch aktualisieren
 Aktualisieren

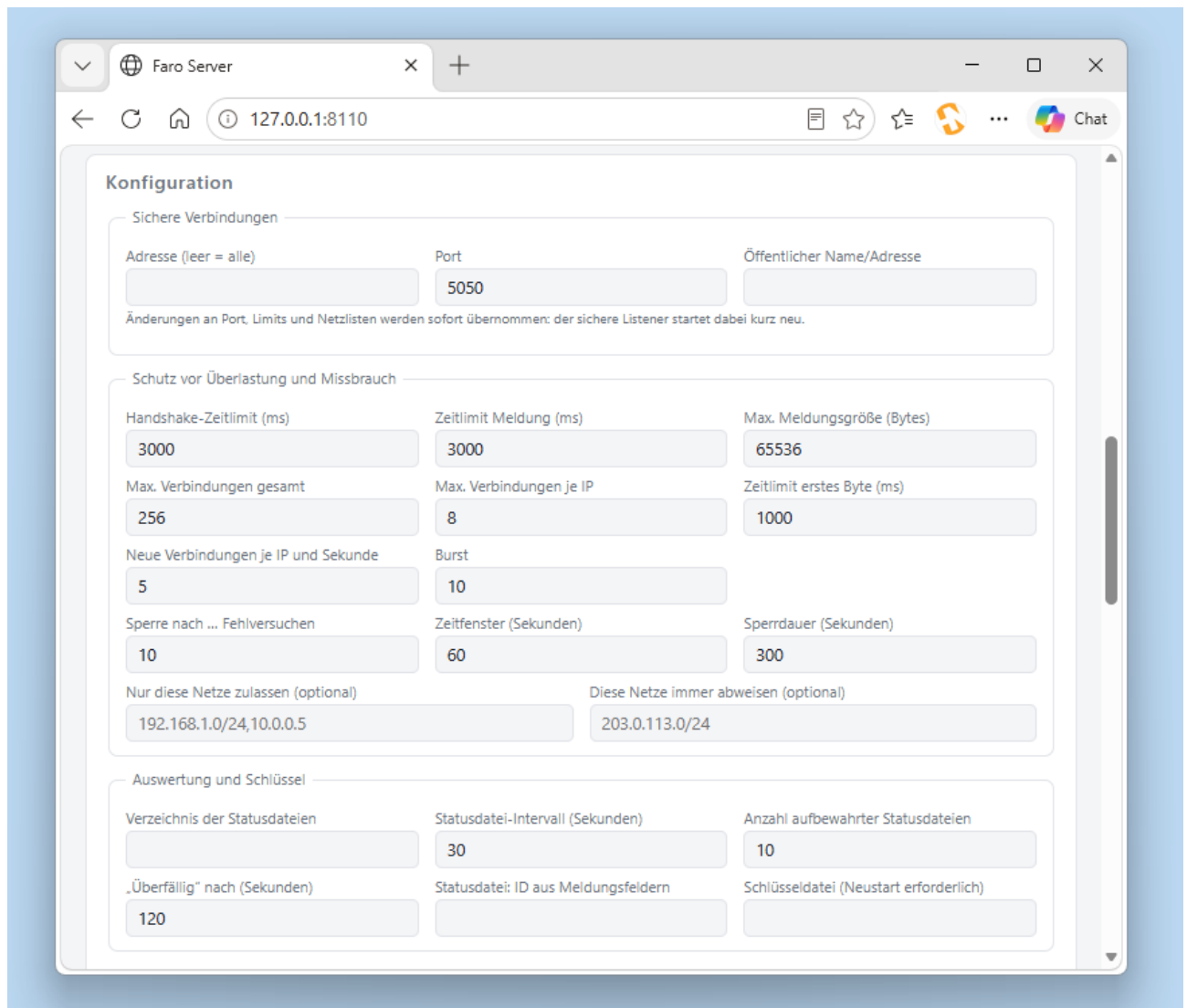
Statusdatei 20261004091312016.accfarostatus (4.10.2026, 11:13:12, 3 Zeilen) - Stand 11:13:41

Client	Empfangen ▼	Alter	Quelle	Host	Host-UUID	Host-Status	Zeit laut Client	Quittung	Anzahl
<div>Test-Client 02</div> <div> <div>timecode_utc17=20261004091318963</div> <div>host_name=LAPNAU07</div> <div>host_uuid=36317b48-0e53-4626-990a-a6895dd823ca</div> <div>host_status=OK</div> <div>message_uuid=3dc7eaf5-37d4-4c01-9bdb-82a050b9c603</div> </div>	4.10.2026, 11:13:19	22s	192.168.178.80	LAPNAU07	36317b48-0e53-4626-990a-a6895dd823ca	OK	4.10.2026, 11:13:18	OK	34
Test-Client 04	4.10.2026, 11:09:53	3m 48s	192.168.178.80	LAPNAU07	36317b48-0e53-4626-990a-a6895dd823ca	OK	4.10.2026, 11:09:53	OK	1
Test-Client 01 gesperrt	4.10.2026, 11:06:32	7m 9s	192.168.178.80	LAPNAU07	36317b48-0e53-4626-990a-a6895dd823ca	OK	4.10.2026, 11:06:32	OK	21

Hinweis: Der Text der Meldungen stammt von den Clients. Die Weboberfläche zeigt ihn ausschließlich als Text an; enthaltene Steuerzeichen oder HTML-Befehle werden nicht ausgeführt. Zu lange Felder werden gekürzt dargestellt (die Statusdatei enthält die vollständige Meldung).

2.6 Konfiguration des Servers

Der Bereich **Konfiguration** der Weboberfläche enthält alle Einstellungen der Datei AccFaroServer.cfg. Änderungen werden mit **Speichern** sofort übernommen; ändern sich Port, Schutzgrenzen oder Netzlisten, startet der Server seinen Listener für sichere Verbindungen selbstständig kurz neu. Nur die Schlüsseldatei und die Einstellungen der Weboberfläche selbst (Port, Adresse) wirken erst nach einem Neustart.



Sichere Verbindungen

Einstellung	Standard	Bedeutung
listenaddress	leer	Adresse, auf der gelauscht wird. Leer = alle Netzwerkadapter (IPv4 und IPv6). Damit lässt sich gezielt das Netzwerksegment wählen, auf dem Meldungen erwartet werden.
listenport	5050	TCP-Port, auf dem die AccFaroClients ihre Meldungen senden.
publichost	Rechnername	Name oder Adresse, unter der die Clients den Server erreichen. Wird in das Client-Paket eingetragen.

Schutz vor Überlastung und Missbrauch (Hintergrund in Kapitel 8)

Einstellung	Standard	Bedeutung
handshaketimeoutms	3000	Gesamtzeit vom Verbindungsaufbau bis zum fertigen Handshake; danach wird ohne Antwort getrennt.
firstbytetimeoutms	1000	Ein echter Client sendet sofort. Verbindungen, die nach dieser Zeit noch nichts gesendet haben, werden getrennt (Schutz vor „Slowloris“).
firstframetimeoutms	3000	Zeit für den Empfang der Meldung nach dem Handshake.
maxframebytes	65536	Maximale Größe einer Meldung. Wird vor dem Lesen geprüft.
maxconnections	256	Gleichzeitig offene Verbindungen insgesamt. Weitere werden sofort geschlossen.
maxperip	8	Gleichzeitige Verbindungen je Absender-IP. Für viele Clients hinter einer gemeinsamen Adresse (NAT) erhöhen.
newconnpersec, newconnburst	5, 10	Neue Verbindungen je Absender-IP: dauerhaft pro Sekunde und kurzzeitig am Stück.
failbanthreshold, failbanwindowsec, failbansec	10, 60, 300	Eine Quelle, die innerhalb von 60 Sekunden 10-mal einen fehlgeschlagenen Handshake verursacht, wird 300 Sekunden gesperrt (0 = nie sperren).
allowcidr,denycidr	leer	Optionale Positiv- bzw. Negativliste von Netzen, kommasetrennt, z. B. 192.168.1.0/24, 10.0.0.5. Ist eine Positivliste angegeben, dürfen nur diese Netze verbinden.

Auswertung und Schlüssel

Einstellung	Standard	Bedeutung
statuspath	status	Verzeichnis für die Statusdateien (bei einer benannten Instanz status.<Instanzname>), relativ zum Programmordner oder absolut.
statusinterval	30	Abstand in Sekunden, in dem jeweils eine neue Statusdatei geschrieben wird.
maxstatusfiles	10	Anzahl der aufbewahrten Statusdateien; ältere werden automatisch gelöscht.
staleafterseconds	120	Ohne Meldung so lange = „überfällig“ (nur Anzeige).
idfromparams	leer	Statusdatei, Spalte 1: ID aus Meldungsfeldern bilden, z. B. so, bq (Abschnitt 5). Leer = Anzeigenname des Clients.
msgoutputdir, msgoutputinterval	leer, 60	Lebenszeichen für AccRoutineMonitor (Kapitel 6): Verzeichnis und Abstand in Sekunden. Leer = aus.
enrollvalidseconds	600	Gültigkeit eines Einmalcodes in Sekunden (Abschnitt 2.3). Nur in der cfg-Datei.
enrollapprovalseconds	300	Zeit, die Sie nach der Anmeldung eines Geräts zum Bestätigen haben. Nur in der cfg-Datei.
enrollmaxfailures	20	Fehlversuche mit falschen Codes, nach denen alle unbenutzten Codes zurückgezogen werden. Nur in der cfg-Datei.
keyfile	leer	Schlüsseldatei. Leer = %ProgramData%\AccMicroServices\AccFaroServer[.Instanz]\clients.keys. Änderung erfordert einen Neustart.
messagefile	message.inf	Vorlage der Quittung (siehe unten).

Protokollierung (logfile, loglevel, logretentiondays, logtoconsole) und Weboberfläche (adminport Standard 8110, adminbindaddress, adminuser, adminpassword, guiautostart, helpink u. a.) entsprechen den übrigen AccMicroServices.

Quittung

Jede Meldung wird vom Server mit einer Quittung beantwortet. Ihr Inhalt stammt aus der Vorlage message.inf (bearbeitbar im Bereich **Quittungsvorlage**). So sieht man direkt am

Client, ob die Verbindung funktioniert und welchen Status der Server hat. In der Vorlage können folgende Platzhalter verwendet werden, die der Server beim Senden durch die aktuellen Werte ersetzt:

Platzhalter ...	... wird ersetzt durch
%message_status%	Ergebnis der Verarbeitung: OK, NO_MESSAGE_UUID (Meldung ohne message_uuid) oder BAD_FRAME (leere Meldung).
%message_uuid%	Die UUID der quittierten Meldung. Der Client prüft sie – eine Quittung mit fehlender oder falscher UUID wird verworfen.
%timecode_utc17%	Uhrzeit des Servers im UTC17-Format (JJJJMMThhmmsssttt).
%host_name%	Name des Server-PCs.
%host_uuid%	Eindeutige Kennung des Server-PCs.
%host_status%	Status des Server-PCs, normalerweise OK.

```
message_status=%message_status%
message_uuid=%message_uuid%
timecode_utc17=%timecode_utc17%
host_name=%host_name%
host_uuid=%host_uuid%
host_status=%host_status%
```

Die Vorlage darf Escape-Sequenzen wie \n für einen Zeilenumbruch enthalten.

2.7 Erreichbarkeit: Port und Firewall

Damit der AccFaroServer Meldungen empfangen kann, muss er von den Clients über eine – ggf. öffentliche – IP-Adresse erreichbar sein. Der frei wählbare **TCP-Port** `listenport` muss von dieser Adresse bis zum AccFaroServer durchgeroutet und an allen Firewalls freigegeben werden, letztlich auch an der Firewall des Betriebssystems, unter dem der Dienst läuft. Genau dieser Port (und der Name bzw. die Adresse aus `publichost`) wird im Client-Paket eingetragen.

Die Clients bauen die Verbindung selbst auf; auf den überwachten Systemen müssen daher **keine eingehenden Ports** geöffnet werden. Eine Namensauflösung (DNS) ist nur nötig, wenn die Clients den Server über einen Namen ansprechen – ein Eintrag der IP-Adresse im Client-Paket (`publichost`) funktioniert ebenso und macht AccFaro unabhängig vom DNS (Kapitel 11).

2.8 Als Windows-Dienst installieren

Für den Dauerbetrieb läuft der Server als Windows-Dienst. Dadurch wird AccFaro bei jedem Systemstart automatisch mitgestartet und läuft auch, ohne dass sich ein Benutzer am System anmeldet. Am einfachsten geht das im **AccMicroServiceKonfigurator** (Kapitel 4): Dienst `AccFaroServer` wählen und **Installieren + Starten** klicken. Alternativ in einer Eingabeaufforderung mit Administratorrechten im Programmordner:

```
AccFaroServer.exe install
AccFaroServer.exe start
```

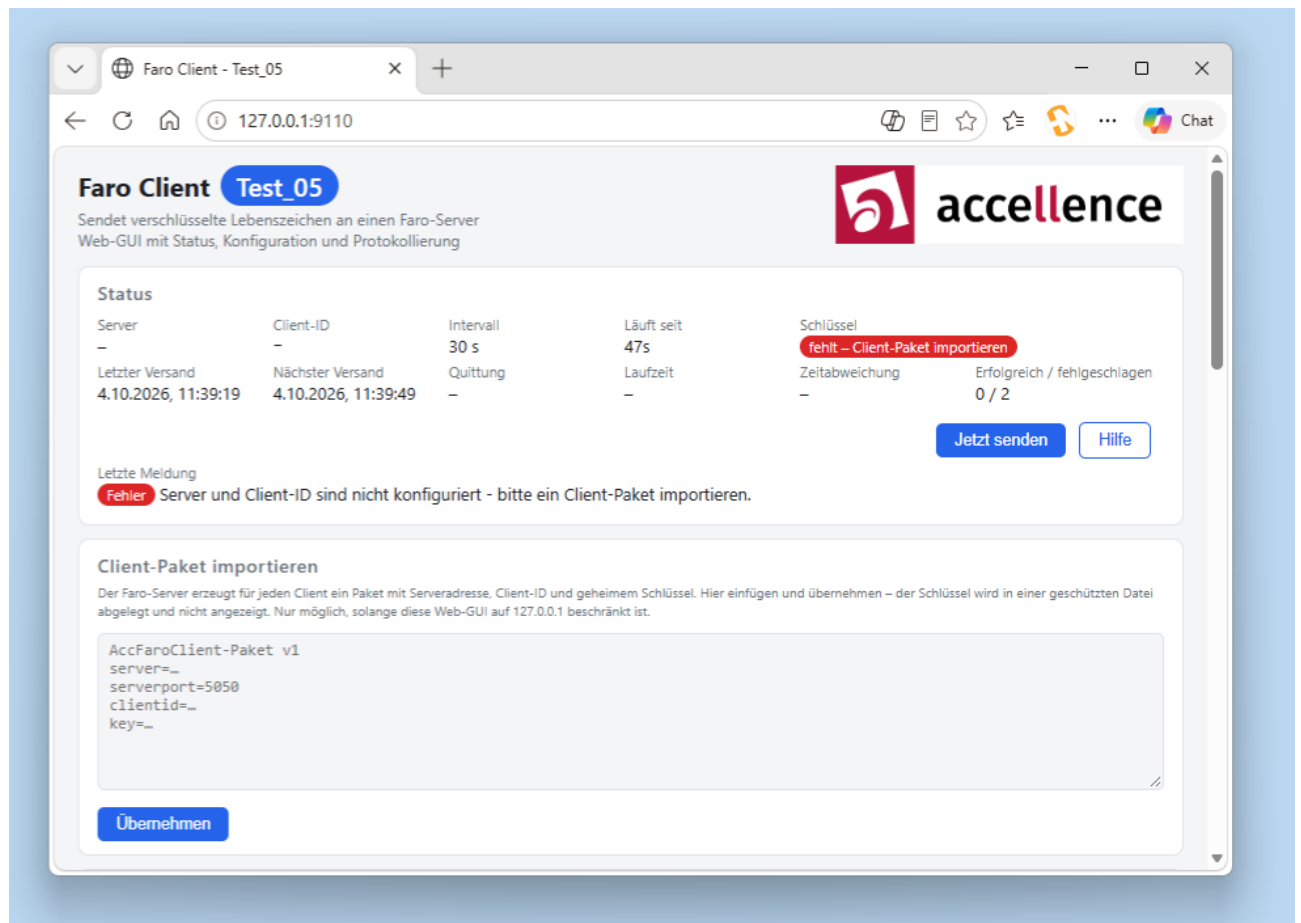
`AccFaroServer.exe stop` beendet den Dienst, `AccFaroServer.exe uninstall` entfernt ihn wieder. Der Dienst läuft standardmäßig unter dem Konto `LocalSystem`; die Schlüsseldatei ist für dieses Konto lesbar.

3 Der AccFaroClient auf den überwachten Systemen

Den AccFaroClient finden Sie auf dem überwachten System im Verzeichnis

C:\EBUES\AccMicroServices\AccFaroClient

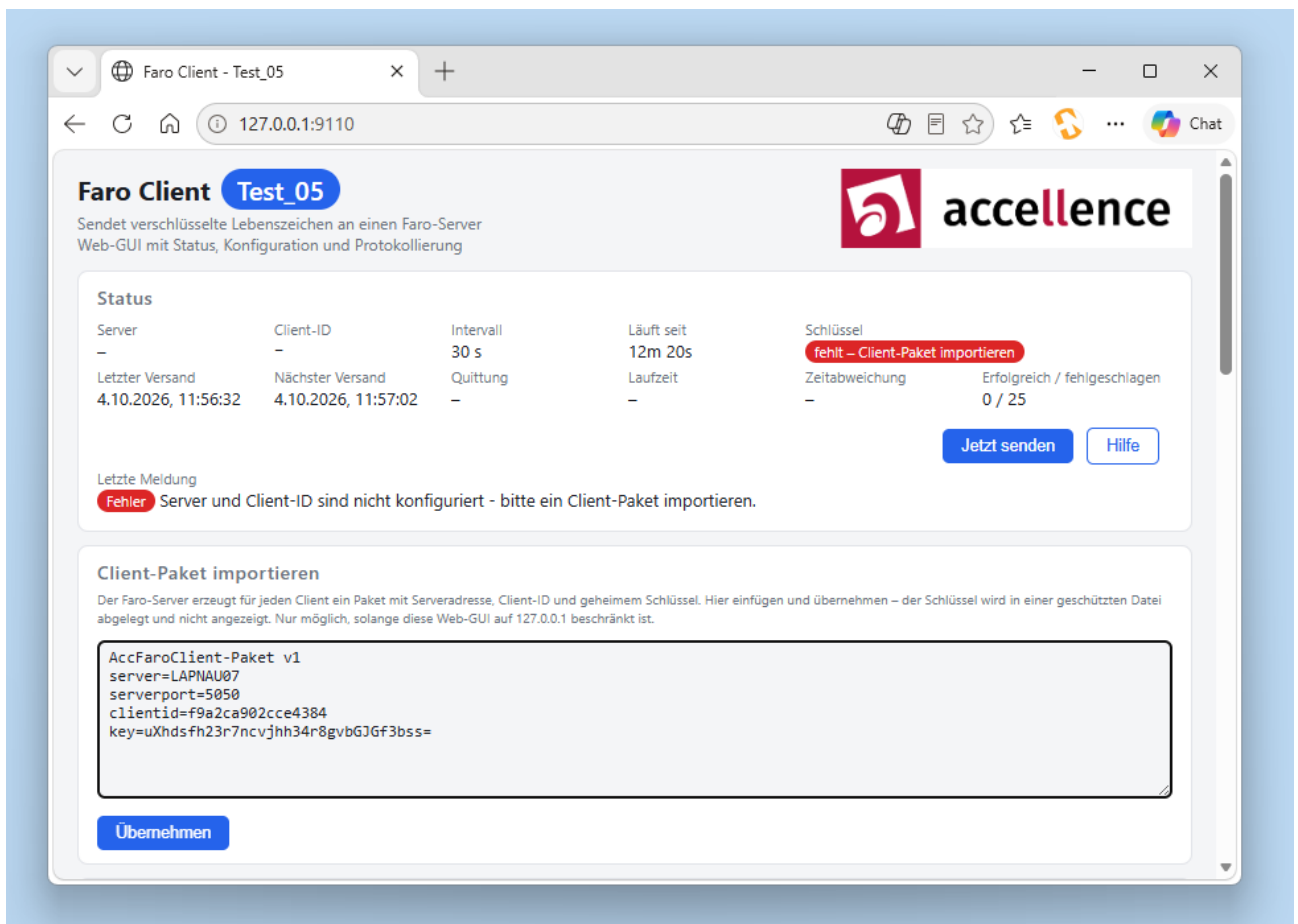
Wie beim Server liegen dort die Anwendung (AccFaroClient.exe), die Konfiguration (AccFaroClient.cfg) und die Vorlage der Meldung (message.inf). Führen Sie AccFaroClient.exe durch Doppelklick aus: Es öffnet sich ein Konsolenfenster, und – bei guiautostart=1 – die Weboberfläche des Clients im Browser (Standard: <http://127.0.0.1:8111/>).



3.1 Client-Paket importieren

Server, Port, Client-ID und Schlüssel werden in einem Schritt aus dem **Client-Paket** übernommen, das der Server beim Anlegen des Clients erzeugt hat (Abschnitt 2.2):

1. Im Bereich **Client-Paket importieren** der Client-Weboberfläche den Paketinhalt einfügen.
2. Auf **Übernehmen** klicken.



Der Client trägt Server, Port und Client-ID in `AccFaroClient.cfg` ein, legt den geheimen Schlüssel in einer geschützten Schlüsseldatei ab und sendet sofort eine erste Meldung. Der Schlüssel erscheint **nie** in der Konfigurationsdatei und wird **nie** angezeigt:

`C:\ProgramData\AccMicroServices\AccFaroClient\client.key`

Hinweis: Der Import ist – wie das Anlegen auf dem Server – nur möglich, solange die Weboberfläche des Clients auf 127.0.0.1 beschränkt ist, damit der Schlüssel nicht unverschlüsselt über das Netz geht. Sie können die Schlüsseldatei auch von Hand auf dem System ablegen; sie enthält den Schlüssel als Base64-Text.

3.2 Alternative: Anmelden mit Einmalcode

Wenn sich das Client-Paket nicht übertragen lässt (Kamera, Recorder, Alarmsystem ohne E-Mail und USB), meldet sich der Client mit einem **Einmalcode** der Leitstelle an. Die Leitstelle erzeugt den Code (Abschnitt 2.3) und nennt Ihnen am Telefon **Server, Port und die 16 Ziffern**:

1. In der Weboberfläche des Clients im Bereich **Anmelden mit Einmalcode Server, Port und Einmalcode** eingeben und auf **Anmelden** klicken. Leerzeichen und Bindestriche im Code spielen keine Rolle; die letzte Ziffer ist eine Prüfziffer, die Tippfehler sofort erkennt.
2. Der Client verbindet sich und zeigt eine große **Bestätigungszahl** (4 Ziffern). **Nennen Sie diese Zahl der Leitstelle am Telefon** und warten Sie.
3. Sobald die Leitstelle die Zahl verglichen und bestätigt hat, erscheint **Angemeldet**: Der Client hat seinen Schlüssel erhalten, in einer geschützten Datei abgelegt (er wird nicht angezeigt) und Server, Port und Client-ID in die Konfiguration übernommen. Die erste Meldung wird sofort gesendet.

Für Geräte **ohne Weboberfläche** gibt es die Kommandozeile (Instanzen wie üblich mit `--instance=NAME`):

```
AccFaroClient.exe enroll --server=leitstelle.example --port=5050 --code="4821 0937 5516 2084"
```

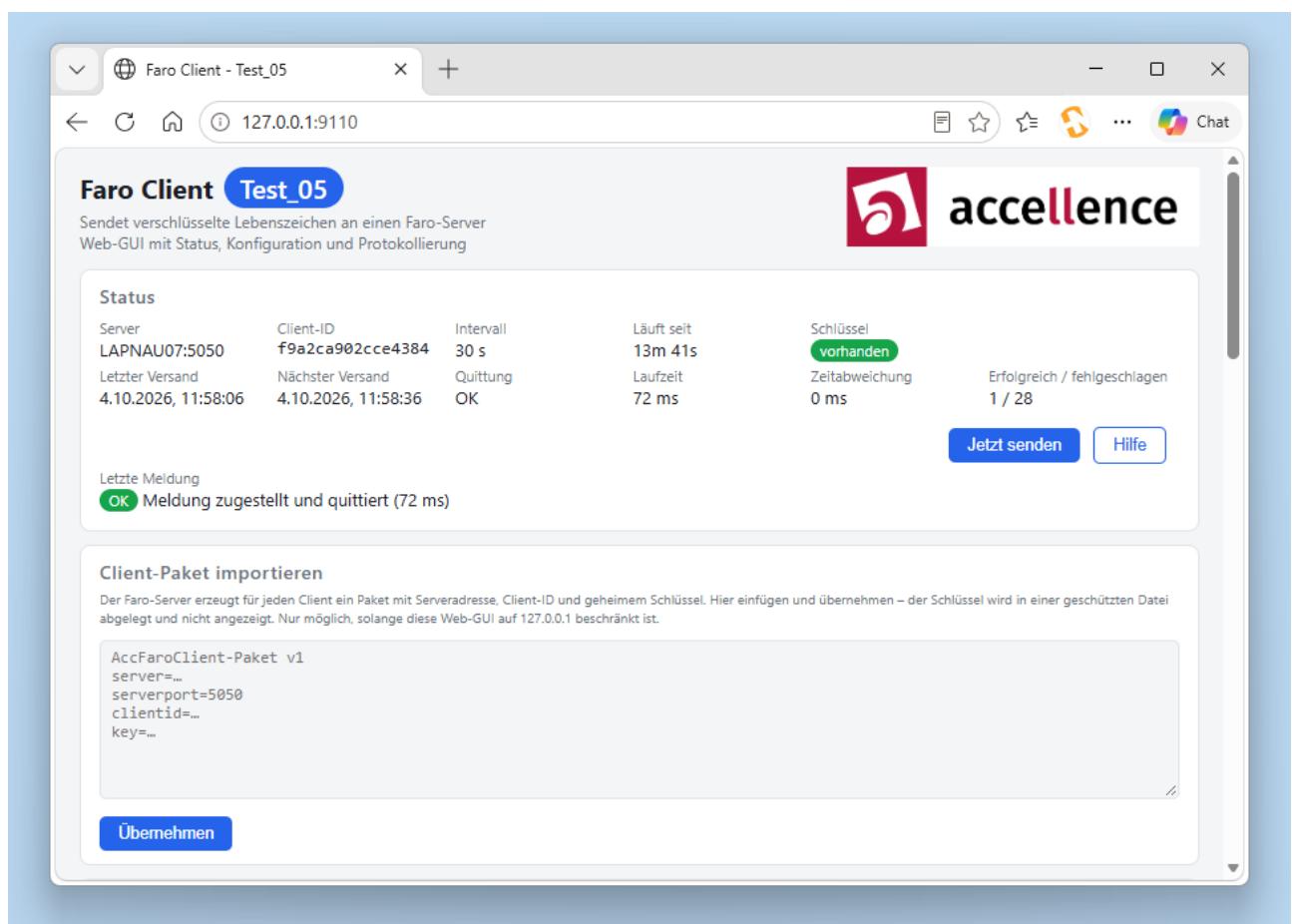
Fehlt `--code`, fragt das Programm danach. Es zeigt die Bestätigungszahl an, wartet auf die Leitstelle und beendet sich mit dem Exit-Code 0 (angemeldet), 1 (abgelehnt, abgelaufen oder fehlgeschlagen) oder 2 (Eingabe fehlerhaft). Der Dienst nimmt die neue Konfiguration und den Schlüssel beim nächsten Zyklus selbst auf.

Hinweis: Ein **falscher, abgelaufener oder bereits benutzter Code** sieht für den Client genauso aus wie ein nicht erreichbarer Server („Anmeldung nicht möglich ...“) – so verrät der Server keine gültigen Codes. Prüfen Sie Code, Server und Port, und lassen Sie sich bei Bedarf einen neuen Code geben. In der Weboberfläche des Clients ist die Anmeldung wie der Paket-Import nur möglich, solange sie auf 127.0.0.1 beschränkt ist; die Kommandozeile ist davon nicht betroffen.

3.3 Status und Test

Im oberen Bereich der Weboberfläche zeigt der Client den Zustand der letzten Meldung:

Anzeige	Bedeutung
Server, Client-ID	Ziel und Identität dieses Clients.
Schlüssel	vorhanden oder fehlt – Client-Paket importieren.
Letzter / Nächster Versand	Zeitpunkt der letzten Meldung und der nächsten geplanten Meldung.
Quittung	Status aus der Antwort des Servers (message_status), normalerweise OK.
Laufzeit	Dauer von Verbindungsaufbau bis Quittung in Millisekunden – ein Maß für die Qualität der Netzwerkverbindung.
Zeitabweichung	Serverzeit minus eigene Zeit beim Empfang der Quittung (grob, enthält die halbe Laufzeit). Nur zur Information: Die Sicherheit hängt nicht von den Uhren ab. Sind alle PCs z. B. per NTP synchronisiert, zeigt der Wert die Netzwerklaufzeit.
Erfolgreich / fehlgeschlagen	Anzahl der Meldungen seit dem Start des Clients.
Letzte Meldung	Ergebnis im Klartext, bei Fehlern mit Ursache (Kapitel 9).



Die Schaltfläche **Jetzt senden** löst sofort eine zusätzliche Meldung aus, unabhängig vom Intervall – ideal zum Testen nach Änderungen.

3.4 Inhalt der Meldung

Der Inhalt der Meldung steht in der Datei `message.inf` (im Bereich **Meldungsvorlage** der Weboberfläche bearbeitbar): eine Zeile je Wert in der Form `schluesseel=wert`. Fehlt `message_uuid`, ergänzt der Client es automatisch. Beispiel für ein System mit EBÜS-Bildquelle:

```
timecode_utc17=%timecode_utc17%
message_uuid=%message_uuid%
host_name=%host_name%
host_uuid=%host_uuid%
tcpip_alarm_port=13003
so=Meyer & Söhne GmbH
bq=Recorder 01
bqt=Geutebrück GCore_v8.2
serial_nr=GB0815-1000
```

Feld	Bedeutung
timecode_utc17	Sendezeitpunkt. Sind die Uhren aller Systeme korrekt synchronisiert, lässt sich damit die Laufzeit der Meldung messen.
message_uuid	Eindeutige UUID zur Nachverfolgung jeder Meldung. Die Quittung des Servers muss dieselbe UUID enthalten.
host_name, host_uuid	Name des PCs und eine eindeutige Kennung für jeden PC.
tcpip_alarm_port	Port, zu dem das überwachte System seine Alarmer per TCP/IP an die Leitstelle sendet.
so, bq, bqt	Name des in EBÜS konfigurierten Schutzobjektes, der Bildquelle und der Bildquellentyp.
serial_nr	Seriennummer zur eindeutigen Identifizierung des Systems.

Weitere Felder können in zusätzlichen Zeilen ergänzt werden; die Feldnamen müssen eindeutig sein. Die Datei wird vor jeder Meldung neu eingelesen, Änderungen wirken daher bei der nächsten Meldung. Folgende Platzhalter ersetzt der Client beim Senden automatisch durch die aktuellen Werte:

Platzhalter ...	... wird ersetzt durch
%timecode_utc17%	Uhrzeit im UTC17-Format (JJJJMMTTThhmmsssttt, www.ebues.de/glossar#UTC17).
%message_uuid%	Eindeutige UUID zur Verwaltung jeder Meldung.
%host_name%	Hostname des PCs, von dem die Meldung verschickt wird.
%host_uuid%	Eindeutige Kennung für jeden PC.
%host_status%	Status dieses PCs, normalerweise OK.

Statt der Datei lässt sich in `AccFaroClient.cfg` auch eine einzeilige Vorlage mit Escape-Sequenzen angeben, z. B. `message=host_name=%host_name%\nmessage_uuid=%message_uuid%`; sie wird nur verwendet, wenn die Nachrichtendatei fehlt oder leer ist.

3.5 Konfiguration des Clients

Alle Einstellungen der Datei `AccFaroClient.cfg` lassen sich im Bereich **Konfiguration** der Weboberfläche bearbeiten und wirken nach dem nächsten Sendeintervall – ein Neustart ist nicht nötig (Ausnahme: Port und Adresse der Weboberfläche).

Einstellung	Standard	Bedeutung
server	–	Name oder IP-Adresse des AccFaroServers (aus dem Client-Paket).
serverport	5050	Port des Servers (aus dem Client-Paket).
clientid	–	ID dieses Clients (aus dem Client-Paket).
keyfile	leer	Schlüsseldatei. Leer = %ProgramData%\AccMicroServices\AccFaroClient[.Instanz]\client.key.
interval	30	Sendeintervall in Sekunden.
connecttimeoutms, handshake timeoutms, replytimeoutms	5000	Zeitlimits in Millisekunden für TCP-Verbindungsaufbau, Handshake und Wartezeit auf die Quittung.
messagefile	message.inf	Datei mit der Vorlage der Meldung.
msgoutputdir, msgoutputinterval	leer, 60	Lebenszeichen für AccRoutineMonitor (Kapitel 6): Verzeichnis und Abstand in Sekunden. Leer = aus.

Protokollierung und Weboberfläche (adminport Standard 8111) entsprechen den übrigen AccMicroServices.

3.6 Als Windows-Dienst installieren

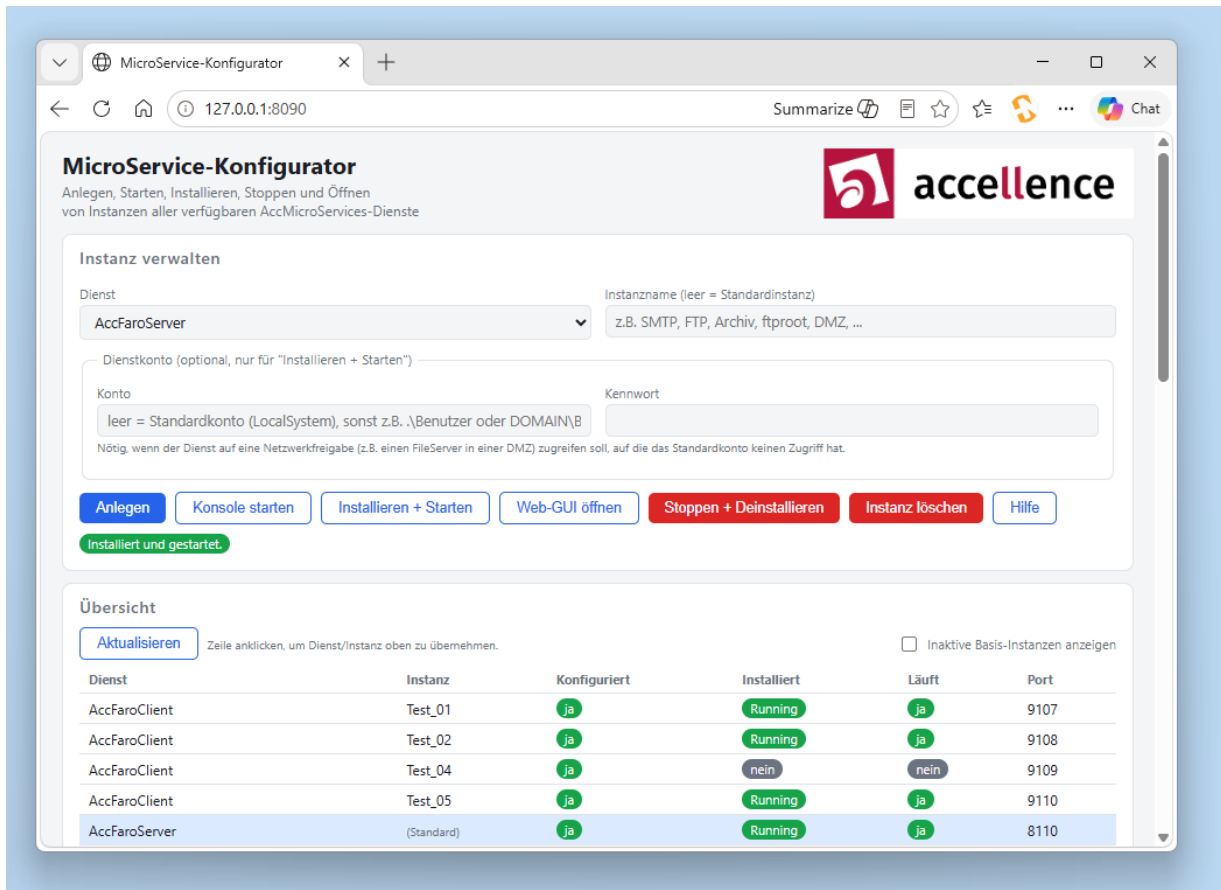
Wie der Server wird auch der Client im **AccMicroServiceKonfigurator** mit **Installieren + Starten** als Windows-Dienst eingerichtet (Kapitel 4) oder in einer Eingabeaufforderung mit Administratorrechten:

```
AccFaroClient.exe install
AccFaroClient.exe start
```

Der Dienst startet mit dem System und läuft ohne angemeldeten Benutzer. Läuft der Dienst nicht unter dem Standardkonto LocalSystem, sondern unter einem eigenen Benutzerkonto, muss dieses Konto die Schlüsseldatei lesen können: Importieren Sie das Client-Paket dann mit diesem Konto oder passen Sie die Zugriffsrechte der Datei entsprechend an.

4 Verwaltung mit dem AccMicroServiceKonfigurator

AccFaroServer und AccFaroClient erscheinen im **AccMicroServiceKonfigurator** automatisch in der Liste **Dienst** und in der **Übersicht** – ohne weitere Einrichtung. Dort lassen sie sich wie alle AccMicroServices anlegen, in der Konsole starten, als Windows-Dienst installieren und stoppen oder deinstallieren, ihre Weboberfläche und diese Dokumentation öffnen (Details siehe AccMicroServices.docx).



Besonderheiten bei AccFaro:

- **Zwei Ports je Server-Instanz:** Neben dem Port der Weboberfläche (adminport) öffnet der Server den Port für die Clients (listenport). Legt der Konfigurator eine **benannte Instanz** an, vergibt er für beide automatisch freie Ports aus seinem Portbereich. Passen Sie den Port für die Clients bei Bedarf in der Weboberfläche an (z. B. auf 5050) und geben Sie ihn in der Firewall frei; das Client-Paket enthält jeweils den aktuellen Wert.
- **Mehrere Instanzen:** Mit benannten Instanzen betreiben Sie z. B. je Kunde oder Netzwerk einen eigenen Server (eigene Clients, eigene Schlüsseldatei, eigenes Statusverzeichnis) beziehungsweise auf einem überwachten System einen Client je Zielsystem (siehe Kapitel 10).
- **Instanz löschen:** Beim Löschen einer benannten Instanz entfernt der Konfigurator auch deren Schlüsselverzeichnis (%ProgramData%\AccMicroServices\AccFaroServer.<Instanz>). Alle zugehörigen Clients verlieren damit ihren Zugang. Die unbenannte Basis-Instanz kann nicht gelöscht werden.

5 Format der Statusdateien

Der AccFaroServer empfängt regelmäßig Meldungen von vielen AccFaroClients. Um stets den aktuellen Zustand aller Systeme bereitzustellen und zugleich den Speicherplatz zu begrenzen, legt er die Daten in **Statusdateien** rollierend ab (mit automatischer „Rotation“). So stehen immer die letzten Statusstände zur Verfügung, ältere werden automatisch gelöscht.

Wie funktioniert die rollierende Speicherung?

- In regelmäßigen Abständen (`statusinterval`, z. B. alle 30 Sekunden) schreibt der Server eine neue Statusdatei in das konfigurierte Verzeichnis (`statuspath`).
- Sobald die Anzahl der Statusdateien `maxstatusfiles` überschreitet, werden die ältesten automatisch gelöscht.
- Zur eindeutigen Erkennung enthält der Dateiname den Zeitpunkt, zu dem die Datei gespeichert wurde, im UTC17-Format, und die Endung `.accfarostatus`, z. B. `20251209214648030.accfarostatus`.
- Jede Statusdatei enthält, zeilenweise gegliedert nach ID, die zuletzt mit dieser ID empfangene Meldung, jeweils mit IP-Adresse und Zeitstempel.

Vorteile: Die Auswertung kann immer auf die letzten, konsistenten Statusstände zugreifen – am sichersten auf die **zweitneueste** Datei, die garantiert vollständig geschrieben ist. Die Dateien bleiben lange genug erhalten, um sie sicher auswerten zu können, und der Speicherplatz bleibt begrenzt.

Beispiel für den Inhalt einer Statusdatei (hier mit `idfromparams=so,bq`; TAB steht für ein Tabulatorzeichen):

```
Meyer & Söhne GmbH,Recorder 01 TAB 192.168.1.101 TAB 20251209174634001 TAB  
timecode_utc17=20251209174634001\nmessage_uuid=...\ntcpip_alarm_port=13003\nso=Meyer & Söhne  
GmbH\nbq=Recorder 01\nnbqt=Geutebrück GCore_V8.2\nserial_nr=GB0815-1000  
Alpha AG,Recorder 02 TAB 192.168.1.102 TAB 20251209174634123 TAB  
timecode_utc17=20251209174634123\nmessage_uuid=...\ntcpip_alarm_port=13004\nso=Alpha  
AG\nbq=Recorder 02\nnbqt=Geutebrück GCore_V8.2\nserial_nr=GB0815-1001
```

Jede Zeile enthält, durch Tabulator getrennt:

1. die **ID** des Systems (siehe unten),
2. die **IP-Adresse**, von der die Meldung dieses Systems empfangen wurde,
3. den **Empfangszeitpunkt** der letzten Meldung im UTC17-Format,
4. die **Meldung** selbst (mehrzeilig; `\n` steht dabei für einen Zeilenumbruch, `\t` für ein Tabulatorzeichen, `\\` für einen Backslash).

Die Datei kann einfach von anderen Programmen ausgewertet werden. In jeder neuen Statusdatei stehen die jeweils letzten Meldungen aller Systeme, von denen seit dem Start des Servers Meldungen empfangen wurden.

Welche ID steht in Spalte 1? Standardmäßig der **Anzeigename**, den Sie beim Anlegen des Clients vergeben haben – er steht fest im Server und kann von einem Client nicht verändert werden. Möchten Sie die ID wie bisher aus Feldern der Meldung bilden (z. B. `so` und `bq` für EBÜS-Bildquellen), tragen Sie die Feldnamen kommasetrennt bei `idfromparams` ein, z. B. `idfromparams=so,bq`. Fehlt ein Feld in einer Meldung, wird ersatzweise der Anzeigename verwendet.

Hinweis: Felder der Meldung stammen vom Client selbst. Ein **kompromittierter** Client könnte darin eine fremde ID angeben und so einen anderen Eintrag überschreiben. Wo das

eine Rolle spielt, lassen Sie `idfromparams` leer und vergeben die Anzeigenamen so, dass Ihre Auswertung sie verwenden kann.

6 Erkennen und Melden von Systemausfällen

Der AccFaroServer dokumentiert für jedes System anhand der ID die aktuelle IP-Adresse, von der die Meldung empfangen wurde, den exakten Zeitpunkt der letzten Meldung sowie alle weiteren Felder, die in der Datei `message.inf` des jeweiligen AccFaroClients konfiguriert sind.

Über die Zeitstempel lassen sich Ausfälle einzelner Systeme oder ganzer Netzsegmente zuverlässig erkennen und dokumentieren: Bleibt die Meldung eines Systems länger als erwartet aus, ist es oder die Netzwerkverbindung zu ihm gestört. In diesem Fall kann die Leitstelle entsprechende Alarme generieren und alle nötigen Maßnahmen einleiten. Die Weboberfläche des Servers zeigt solche Clients bereits als **überfällig** an.

6.1 Lebenszeichen für AccRoutineMonitor

Damit auch der **Faro-Dienst selbst** überwacht werden kann, schreiben AccFaroServer und AccFaroClient auf Wunsch ein regelmäßiges **Lebenszeichen** in einen Ordner, den das Überwachungswerkzeug **AccRoutineMonitor** beobachtet. Dazu tragen Sie in der Weboberfläche (Bereich **Konfiguration**, Gruppe **Lebenszeichen für AccRoutineMonitor**) oder in der `cfg`-Datei ein **Meldungsverzeichnis** (`msgoutputdir`) und den **Abstand** in Sekunden (`msgoutputinterval`, Standard 60) ein. Ist das Verzeichnis leer, wird nichts geschrieben.

In diesem Abstand entsteht eine kleine Datei mit dem Namen `<Kennung>_<UTC17-zeitstempel>.msg` (z. B. `AccFaroServer_20261004123015123.msg`; bei einer benannten Instanz mit deren Namen). Der Zeitstempel im Dateinamen ist wichtig: AccRoutineMonitor erkennt eine Datei nur dann als neu, wenn sich der Name ändert. Der Inhalt hat ein von AccRoutineMonitor fest vorgegebenes Format:

```
RoutineMessage
InstanceName=<Kennung>
TimeStamp=<UTC17-Zeitstempel>
IntervalTime=<msgoutputinterval>
--- eof ---
```

- **Der AccFaroServer** schreibt das Lebenszeichen **nur, solange sein sicherer Listener läuft**, also solange er Verbindungen der Clients annehmen kann. Kann der Port nicht geöffnet werden (z. B. weil er belegt ist), bleibt die Datei aus, und AccRoutineMonitor schlägt Alarm; sobald der Server den Port wieder öffnen konnte (Wiederholung alle 10 Sekunden), erscheinen die Dateien wieder.
- **Der AccFaroClient** schreibt das Lebenszeichen, **solange er läuft** – unabhängig davon, ob der Server gerade erreichbar ist. Ob die Meldungen ankommen, zeigt der Server selbst (Zustand „überfällig“, Kapitel 2).

Hinweis: Die Dateien sammeln sich im Meldungsverzeichnis an. Wie bei AccFileCleaner und AccFileMover räumt eine AccFileCleaner-Instanz, die auf dieses Verzeichnis zeigt, sie wieder auf.

7 Überwachung der Alarmwege

Indem die AccFaroClients zum Senden ihrer Meldungen exakt dieselben Netzwerkverbindungen zur Leitstelle nutzen wie das damit überwachte System, werden auch dessen Alarmwege überwacht: Kommt die Meldung an, funktioniert auch der Weg, den Alarme nehmen.

8 Datensicherheit

Die Verbindung zwischen AccFaroClient und AccFaroServer nutzt die gemeinsame Sicherheitsschicht der AccMicroServices. Sie ist bewusst auf bewährte Standards aufgebaut, statt eigene Verfahren zu erfinden:

Schutzziel	Umsetzung
Vertraulichkeit und Integrität	Alle übertragenen Daten werden mit TLS 1.3 und AES-256-GCM verschlüsselt und gegen Veränderung gesichert. Ältere TLS-Versionen werden abgelehnt.
Authentifizierung	Jeder Client hat eine eigene ID und einen eigenen zufälligen 256-Bit-Schlüssel (Pre-Shared Key). Beide Seiten beweisen im Handshake den Besitz dieses Schlüssels – ein falscher Server wird vom Client ebenso erkannt wie ein falscher Client vom Server, ganz ohne Zertifikatsverwaltung.
Schutz vor Wiedereinspielen (Replay)	Jede Verbindung beginnt mit einem frischen Handshake mit neuen Zufallswerten; ein mitgeschnittener Datenverkehr lässt sich nicht wiederholen. Die Sicherheit hängt nicht von den Uhren der Systeme ab – Zeitstempel in den Meldungen dienen nur der Information. Die Quittung des Servers ist über die <code>message_uuid</code> an die gesendete Meldung gebunden.
Vorwärtsgeheimnis (Forward Secrecy)	Der Sitzungsschlüssel entsteht durch einen Schlüsselaustausch (ECDHE) pro Verbindung. Wird ein Schlüssel später bekannt, lässt sich der damals mitgeschnittene Verkehr trotzdem nicht entschlüsseln.
Einzelne Clients sperren	Ein Client lässt sich im Server sofort sperren oder löschen. Ein gesperrter oder gelöschter Zugang, ein unbekannter Client und ein falscher Schlüssel sind von außen nicht zu unterscheiden – ein Angreifer kann gültige Client-IDs nicht durch Probieren herausfinden.
Anmeldung mit Einmalcode	Einmaliger, 10 Minuten gültiger 16-stelliger Code mit Prüfziffer; der Schlüssel entsteht erst nach der Bestätigung durch den Administrator (Bestätigungszahl aus der verschlüsselten Verbindung); der Code wird nie gespeichert oder protokolliert und nach vielen Fehlversuchen zurückgezogen.

Schutz vor Überlastung und Angriffen: Unzulässige Verbindungen werden mit geringstmöglichem Ressourcenverbrauch **ohne Antwort** beendet, **bevor** Verschlüsselungsaufwand entsteht:

- Verbindungen, die den Handshake nicht rechtzeitig abschließen (z. B. langsame „Slowloris“-Angriffe) oder keine gültigen Daten senden, werden nach wenigen Sekunden getrennt.
- Die Zahl gleichzeitiger Verbindungen ist insgesamt und je Absender-IP begrenzt, ebenso die Rate neuer Verbindungen je IP.
- Quellen, die wiederholt Handshakes verfehlen, werden vorübergehend gesperrt; optional lassen sich Netze per Positiv- oder Negativliste freigeben bzw. ausschließen.
- Die Größe einer Meldung ist begrenzt und wird geprüft, **bevor** sie gelesen wird.
- Fehlgeschlagene Verbindungsversuche werden im Protokoll nur zusammengefasst und mengenbegrenzt gemeldet; Angaben aus unbeglaubigten Verbindungen werden bereinigt, damit sich nichts in das Protokoll einschleusen lässt.

Schutz der Weboberflächen: Die Weboberfläche läuft im Browser des Administrators und muss auch dort geschützt werden – eine fremde Webseite, die gerade im selben Browser geöffnet ist, darf sie nicht fernsteuern können. AccFaroServer und AccFaroClient prüfen deshalb den Host-Namen jeder Anfrage (nur 127.0.0.1, localhost und [::1]; Schutz vor „DNS-Rebinding“), lehnen Anfragen fremder Seiten ab und verlangen für jede Änderung ein Sicherheits-Token, das nur die eigene, vom Dienst ausgelieferte Seite kennt. Die Antworten enthalten Sicherheits-Header (kein Einbetten in fremde Seiten, kein Zwischenspeichern). Dateipfade (Log, Nachrichtendatei, Statusverzeichnis) lassen sich über die Weboberfläche nur auf **relative Pfade unterhalb des Programmordners** ändern – die Dienste laufen mit weitreichenden Rechten, ein beliebiger Pfad würde sonst das Überschreiben beliebiger Dateien erlauben. Die Schlüsseldatei und absolute Pfade lassen sich nur direkt in der cfg-Datei einstellen. Der Zugriff auf die Weboberfläche über den AccWebUiGateway ist möglich, das Erzeugen und Importieren von Schlüsseln jedoch nicht.

Hinweis: Die Weboberfläche ist standardmäßig ohne Anmeldung nur vom **selben Rechner** aus erreichbar. Auf einem Rechner, an dem sich mehrere Benutzer anmelden (z. B. Terminalserver), kann deshalb jeder dieser Benutzer die Oberfläche bedienen. Setzen Sie dort adminuser und adminpassword.

Hinweis: Eine **verteilte Überlastung des Netzwerks** (DDoS) kann keine Anwendung allein abwehren – dafür sind Firewall und Netzbetreiber zuständig. AccFaro sorgt dafür, dass der Server selbst bei vielen unzulässigen Verbindungen nicht zusätzlich belastet wird und die zulässigen Clients weiter bedient werden.

Schutz der Schlüssel: Die Schlüssel stehen nicht in Konfigurationsdateien, sondern in eigenen Schlüsseldateien im Ordner %ProgramData%\AccMicroservices\..., auf die nur SYSTEM, die Administratoren und das Dienstkonto zugreifen können. Sie werden in keiner Weboberfläche angezeigt – mit Ausnahme des Client-Pakets, einmalig beim Anlegen, und nur, solange die Weboberfläche auf 127.0.0.1 beschränkt ist. Behandeln Sie die Schlüsseldatei des Servers und Sicherungen davon wie ein Passwort.

Was AccFaro nicht verhindert: Ein Client, dessen Rechner kompromittiert wurde, kann bis zu seiner Sperrung weiter (auch falsche) Meldungen senden – sperren Sie ihn dann sofort. Wird der Server-PC selbst kompromittiert, sind alle seine Client-Schlüssel betroffen; legen Sie dann neue Clients an.

9 Was tun wenn ...

Prüfen Sie zuerst die Statusanzeige und das Log der betroffenen Weboberfläche (Bereich **Log** am Ende der Seite) beziehungsweise die Konsolenausgabe.

Typische Meldungen des **Clients** im Feld **Letzte Meldung**:

Meldung	Ursache und Abhilfe
„Server und Client-ID sind nicht konfiguriert“	Es wurde noch kein Client-Paket importiert (Abschnitt 3.1).
„Schlüsseldatei ... nicht lesbar“ bzw. „enthält keinen gültigen 256-Bit-Schlüssel“	Das Client-Paket fehlt oder wurde unvollständig kopiert. Paket erneut importieren; ist es nicht mehr vorhanden, auf dem Server einen neuen Client anlegen.
„Schlüsselverzeichnis kann nicht abgesichert werden ... gehört einem Benutzer, der weder SYSTEM noch Administrator ist“	Der Dienst startet nicht (Windows meldet beim Starten einen Fehler). Der Ordner %ProgramData%\AccMicroServices (bzw. ein Unterordner) gehört einem normalen Benutzer; AccFaro verwendet solche Ordner nicht, weil ein Benutzer dort die Zugriffsrechte wieder ändern und die Schlüssel lesen könnte. Gehört der Ordner einem Administrator-Konto oder SYSTEM, erscheint die Meldung nicht. Abhilfe: In einer Eingabeaufforderung als Administrator takeown /F "%ProgramData%\AccMicroServices" /A ausführen (Besitzer = Administratoren) und den Dienst erneut starten – oder den Ordner löschen, wenn er nur Testdaten enthält.
„Servername ... konnte nicht aufgelöst werden“	Der Name aus server ist im Netz des Clients unbekannt. Namen prüfen oder in publichost des Servers eine IP-Adresse verwenden.
„Verbindung abgelehnt oder nicht erreichbar“ / „Zeitüberschreitung beim Verbindungsaufbau“	Der Server läuft nicht, oder Port bzw. Adresse stimmen nicht, oder eine Firewall/ein Router blockiert den Port (Abschnitt 2.7). Prüfen Sie im Server-Status, ob Sichere Verbindungen aktiv sind.
„Handshake fehlgeschlagen ...“	Client-ID oder Schlüssel stimmen nicht, der Client wurde gesperrt oder gelöscht , oder die Quelle wurde wegen vieler Fehlversuche vorübergehend gesperrt bzw. durch eine Zugriffsbegrenzung abgewiesen. Der Server unterscheidet das nach außen bewusst nicht; im Server-Log und im Bereich Clients sehen Sie die Ursache.
„Anmeldung nicht möglich: Handshake fehlgeschlagen ...“ (Anmeldung mit Einmalcode)	Der Code ist falsch, abgelaufen oder schon benutzt , oder Server und Port stimmen nicht bzw. sind nicht erreichbar. Der Client kann das nicht unterscheiden (Absicht). Eingaben prüfen, bei Bedarf einen neuen Code in der Leitstelle erzeugen lassen.
„Die Leitstelle hat die Anmeldung abgelehnt“ / „Die Zeit ... ist abgelaufen“	Die Bestätigungszahl stimmte nicht überein oder die Leitstelle hat nicht rechtzeitig bestätigt (enrollapprovalseconds). Neuen Code erzeugen und das Telefonat wiederholen.
„Keine Quittung innerhalb des Zeitlimits“	Das Netz ist sehr langsam oder der Server überlastet. replytimeouts erhöhen; Server-Log prüfen.
„Quittung abgelehnt: message_uuid ...“	Die Quittung gehört nicht zur gesendeten Meldung. Das sollte nie vorkommen und deutet auf eine Manipulation oder einen Fehler der Quittungsvorlage des Servers hin.

Mögliche Probleme des **Servers**:

Beobachtung	Ursache und Abhilfe
Status zeigt „inaktiv“ mit Fehlermeldung zum Port	Der Port listenport ist bereits belegt oder die Adresse ungültig. Anderen Port wählen und speichern – der Server versucht es automatisch erneut.
Ein Client steht auf „überfällig“	Der Client sendet nicht (Dienst gestoppt, System aus, Netz gestört) oder wird abgewiesen. Im Client-Log und im Server-Log nachsehen. Die Frist stellt staleafterseconds ein.
Statusdateien werden nicht geschrieben	statuspath ist nicht beschreibbar (Rechte des Dienstkontos prüfen). Das Server-Log nennt den Pfad.

Beobachtung	Ursache und Abhilfe
Browser meldet „403 Forbidden“ bzw. „Unerwarteter Host-Header“	Die Weboberfläche wurde über einen anderen Namen als 127.0.0.1 oder localhost aufgerufen (z. B. über den Rechnernamen). Rufen Sie sie über <code>http://127.0.0.1:<Port>/</code> auf; für den Zugriff aus dem Netz den AccWebUiGateway verwenden.
„403 Forbidden: Sicherheits-Token fehlt oder ist ungültig“ (in der Weboberfläche)	Die Seite wurde geladen, bevor der Dienst neu gestartet wurde; das Sicherheits-Token gilt nur für einen Dienstprozess. Die Weboberfläche erneuert es beim ersten Fehler selbst und wiederholt die Aktion; bleibt die Meldung bestehen, die Seite mit F5 neu laden.
Änderung eines Pfades wird abgelehnt	Über die Weboberfläche sind nur relative Pfade unterhalb des Programmordners zulässig (Kapitel 8). Absolute Pfade direkt in der <code>cfg</code> -Datei eintragen.
Viele Clients hinter einer Adresse werden abgewiesen	<code>maxperip</code> bzw. <code>newconnpersec/newconnburst</code> sind für die gemeinsame Adresse (NAT) zu niedrig eingestellt.

10 Redundanter Betrieb

Für erhöhte Ausfallsicherheit können mehrere AccFaroClients parallel betrieben werden, die ihre Meldungen über unterschiedliche Netze an verschiedene IP-Adressen senden. Dazu legen Sie auf dem überwachten System im AccMicroServiceKonfigurator je Zielserver eine **benannte Instanz** des Clients an und importieren das jeweilige Client-Paket.

Dank des Parameters `messagefile` in der Konfiguration genügt es, die zu versendenden Daten nur einmal zentral bereitzustellen: Alle Instanzen können auf dieselbe Meldungsdatei zeigen. Damit wird der Datenpflegeaufwand minimiert und die Konsistenz der gemeldeten Daten sichergestellt.

Entsprechend können unter den verschiedenen Ziel-IP-Adressen auch mehrere AccFaroServer betrieben werden, deren Daten im Sinne einer robusten Microservice-Architektur von weiteren Diensten miteinander abgeglichen werden können. Diese Dienste können auf Basis der Daten auch den Ausfall einzelner Systeme oder ganzer Netze zuverlässig erkennen und melden.

11 Digitale Souveränität

Durch **Plattform-Unabhängigkeit** (Windows, Linux) und die **Unabhängigkeit von anfälligen zentralen Diensten** wie dem DNS trägt AccFaro zu mehr digitaler Souveränität bei.

AccFaro

- funktioniert auch bei Manipulationen, Überlastung (DDoS) oder Ausfall des DNS (man denke als Beispiel nur an die Störung bei Cloudflare vom 3.12.2025)
- aktualisiert eine – z. B. nach der täglichen Zwangstrennung – geänderte IP-Adresse wesentlich schneller als das DNS (im ungünstigsten Fall kann es bis zu zwei volle Tage dauern, bis jeder DNS-Server im Internet die alte IP-Adresse vergessen und die Information aktualisiert hat, dass diese IP nicht mehr zu dem System passt).

12 Download und Updates

AccFaroClient und AccFaroServer sind Bestandteil der AccMicroServices und werden mit dem EBÜS-Setup ins Verzeichnis `c:\EBUES\AccMicroServices` installiert. Aktuelle Versionen stehen mit den EBÜS-Setups unter www.ebues.de/setup bereit. Die Versionsnummer finden Sie in der Fußzeile dieses Dokuments und in der Startmeldung der Dienste.

13 Support / Hotline

Weitere Informationen zu AccFaro finden Sie stets aktuell unter www.ipracc.de

Haben Sie noch Fragen oder Wünsche zu AccFaro?

Dann wenden Sie sich bitte

- telefonisch unter 0511 - 277.2490
- per E-Mail an support@accellence.de

an unsere Hotline. Wir sind Werktags von 9:00-17:00 Uhr zu erreichen.

Wir wünschen Ihnen viel Erfolg bei Ihrer Arbeit mit AccFaro und stehen für Ihre Wünsche und Fragen jederzeit gern zu Ihrer Verfügung.