

SIP Web Phone & Extension Manager

(FreeSWITCH + WebRTC)

1. Overview

This system provides:

- A **browser-based SIP softphone** (WebRTC via JsSIP)
 - A **web-based admin page** to manage FreeSWITCH extensions
 - Static **FreeSWITCH XML directory** backend
 - No SIP logic in Python — the backend only:
 - Serves HTML
 - Edits XML
 - Runs `fs_cli -x reloadxml`
-

2. Access URLs

Function	URL
SIP Softphone	<code>https://<domain>/sip</code>
Extension Admin	<code>https://<domain>/sip/admin</code>

A small ⚙ **gear icon** appears at the **top-right of the softphone page** and links to the admin page.

3. SIP Softphone Page

Features

- Register SIP extension via **WSS**
- Make / receive calls between extensions
- Dial pad with **backspace** (⌫)
- Light / Dark theme toggle
- Works with FreeSWITCH `mod_sofia` (WebSocket)

Screenshot ① – SIP Softphone

SIP 웹폰 데모

브라우저에서 바로 테스트용 / simple web test

☀ 라이트 모드

간단 SIP 웹폰

Simple SIP Web Phone

WebSocket 서버 (WS/WSS) Server URL

wss://freeswitch.throw.im:7443

SIP 도메인 SIP Domain

43.202.112.167

내 내선번호 (Extension / SIP user)

1000

비밀번호 Password

.....

등록 / Register

통화 종료 / Call ended

발신할 내선번호 (Number to dial)

1111

1

2

3

4

5

6

7

8

9

*

0

☒

통화 / Call**끊기 / Hang Up**

참고: 이 페이지는 브라우저 → PBX(예: FreeSWITCH)로 WebSocket(SIP over WebSocket)만 사용합니다. Python 백엔드는 HTML만 서빙합니다. (simple static server only)

Purpose

Manage **static XML extensions** located at:

```
/usr/local/freeswitch/conf/vanilla/directory/default/
```

Each extension corresponds to:

```
<extension>.xml
```

Available Actions

- Create extension
- Set / reset password
- Edit caller ID name / number
- Delete extension
- Auto-runs **reloadxml** after every change

Screenshot ② – Extension Manager

SIP WebPhone 소프트폰 / Softphone **내선 관리 / Extensions**

FreeSWITCH 내선 관리 (정적 XML) / FreeSWITCH Extensions (static XML)

내선번호 / Extension: 표시 이름 (선택) / Display name (optional): 발신자 번호 (선택) / Caller ID number (optional): 비밀번호 (선택) / Password (optional):

생성 / Create **새로고침 / Refresh**

Source directory: `/usr/local/freeswitch/conf/vanilla/directory/default`
 After every change, the server runs `fs_cli -x reloadxml`.

Ext	Caller ID Name	Caller ID #	Has Auth	동작 / Actions
1000 11/28/2025, 10:00:15 AM	Extension 1000	1000	yes	비밀번호 재설정 / Reset PW 삭제 / Delete
1001 11/28/2025, 9:30:37 AM	Extension 1001	1001	yes	비밀번호 재설정 / Reset PW 삭제 / Delete
1002 11/28/2025, 9:30:37 AM	Extension 1002	1002	yes	비밀번호 재설정 / Reset PW 삭제 / Delete
1003 11/28/2025, 9:30:37 AM	Extension 1003	1003	yes	비밀번호 재설정 / Reset PW 삭제 / Delete
1004 11/28/2025, 9:30:37 AM	Extension 1004	1004	yes	비밀번호 재설정 / Reset PW 삭제 / Delete
1005 11/28/2025, 9:30:37 AM	Extension 1005	1005	yes	비밀번호 재설정 / Reset PW 삭제 / Delete
1006 11/28/2025, 9:30:37 AM	Extension 1006	1006	yes	비밀번호 재설정 / Reset PW 삭제 / Delete
1007 11/28/2025, 9:30:37 AM	Extension 1007	1007	yes	비밀번호 재설정 / Reset PW 삭제 / Delete
1008 11/28/2025, 9:30:37 AM	Extension 1008	1008	yes	비밀번호 재설정 / Reset PW 삭제 / Delete
1009 11/28/2025, 9:30:37 AM	Extension 1009	1009	yes	비밀번호 재설정 / Reset PW 삭제 / Delete

5. Admin Authentication (Important)

How admin protection works

The admin page and API are protected using **HTTP Basic Auth**.

Authentication is **enabled only if** the environment variable is set:

```
SIP_ADMIN_PASS=<password>
```

Optional:

```
SIP_ADMIN_USER=admin    # default: admin
```

Behavior

Condition	Result
<code>SIP_ADMIN_PASS</code> unset	Admin page is open (dev mode)
<code>SIP_ADMIN_PASS</code> set	Browser prompts for username/password

Browser prompt

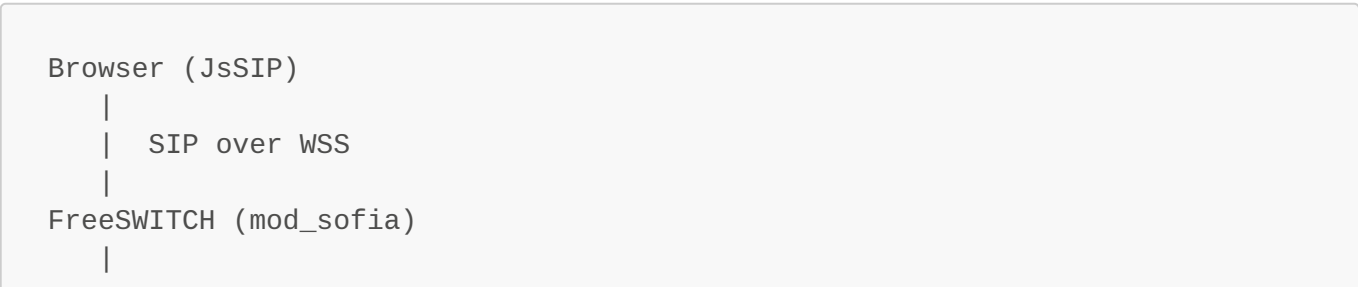
When you first visit `/sip/admin`, the browser shows a login dialog.

- **Username:** `admin` (or `SIP_ADMIN_USER`)
- **Password:** value of `SIP_ADMIN_PASS` (!insunet-fc)

6. Password Handling

- Passwords are stored in **FreeSWITCH XML**
- On creation:
 - If password field is empty → auto-generate
- On reset:
 - You can enter a custom password
 - Or leave blank to auto-generate
- Generated password is shown **once only**

7. Call Flow Summary



```
| RTP / SRTP  
|  
Browser / SIP Phone
```

8. Operational Notes

- If audio becomes one-way:
 - Reload page
 - Re-register
 - Check browser mic permissions
- All extension changes require:
 - Write permission on directory XML
 - Successful `reloadxml`

9. Restart Commands

After code changes:

```
python3 -m py_compile sip_app.py  
sudo systemctl restart sipapp
```

FreeSWITCH reload (automatic via admin):

```
fs_cli -x reloadxml
```