REFERÊNCIA DE SEGURANÇA PARA TI

DWSIM

Documento de referência descrevendo como o DWSIM, as DWSIM Patreon Extensions e o dwsim-assistant interagem com a rede e com o sistema de arquivos local — para uso por equipes de TI corporativas que estão validando estes componentes para instalação em estações gerenciadas.

Data	2026-04-27
Autor	Daniel Wagner Oliveira de Medeiros — autor e desenvolvedor do DWSIM e deste relatório
Componentes	DWSIM (simulador de processos) · DWSIM Patreon Extensions (componentes oferecidos a apoiadores com doação mensal ativa no Patreon) · dwsim-assistant (assistente de IA)
Público	Equipes de TI e Segurança da Informação corporativas
Escopo	Pegada de instalação, endpoints de rede, listeners de entrada, segredos locais, superfície de execução de código, componentes de terceiros, dados que saem da estação, política de firewall recomendada

---

Uma regra de saída "deny by default" mais a allowlist do §9 é suficiente para confinar o DWSIM aos casos de uso que seus usuários precisam.

Sumário

1. Sumário executivo
2. Pegada de instalação
3. Comportamento de rede
4. Listeners de entrada
5. Segredos e credenciais locais
6. Superfície de execução de código
7. Componentes de terceiros
8. Dados que saem da estação
9. Política de firewall e configuração corporativa recomendada

Objetivo: descrever como o DWSIM, as DWSIM Patreon Extensions e o dwsim-assistant interagem com a rede e com o sistema de arquivos local, para que a TI corporativa valide que é seguro instalá-los e executá-los em estações gerenciadas. **Público:** equipes de TI / Segurança da Informação corporativas. **Autor:** Daniel Wagner Oliveira de Medeiros — autor e desenvolvedor do DWSIM e deste relatório. **Data:** 2026-04-27.

> **Conclusão direta:**
>
> - **DWSIM core** é um simulador de processos desktop. O acesso à rede de saída é opcional; a pegada de instalação se limita ao perfil do usuário e a uma única chave de Registro sob `HKCU`. Sem serviços em background, sem canal de auto-update.
> - **DWSIM Patreon Extensions** são componentes adicionais disponibilizados apenas a apoiadores com doação mensal recorrente ativa no Patreon. Adicionam operações unitárias extras e um convergence assistant opcional; o convergence assistant envia parâmetros de simulação anonimizados e pode ser desligado nas configurações do usuário.
> - **dwsim-assistant** é um assistente de IA opcional empacotado como `.exe` único. Faz bind apenas em localhost, é protegido por um token por execução, busca suas credenciais de nuvem em runtime apenas em memória (nada em disco), mantém telemetria desligada por padrão e recusa-se a executar IronPython sugerido pela IA a menos que o usuário habilite explicitamente.

---

1. Sumário executivo

Componente	Saída de rede	Listeners de entrada	Segredos locais	Execução de código	Recomendação
DWSIM core	🟢 Opcional (Simulate365, verificação de licença, consultas químicas manuais)	🟢 Nenhum por padrão	🟢 Tokens OAuth em `HKCU\SOFTWARE\DWSIM` apenas após login do usuário	🟡 IronPython in-process opcional para scripts anexados a objetos de fluxograma	**Permitir**
DWSIM Patreon Extensions	🟡 Upload anonimizado do convergence assistant quando habilitado	🟢 Nenhum no cliente	🟢 Sem segredos no cliente	🟢 Nenhum além do core	**Permitir**
dwsim-assistant	🟡 Provedor LLM, logging Supabase opcional, leitura OPC/PI opcional	🟢 Apenas `127.0.0.1`, todos os endpoints protegidos por token	🟢 Sem segredos em disco	🟢 Execução de IronPython desligada por padrão	**Permitir**

Legenda: 🟢 aceitável · 🟡 revisar e aplicar a política de firewall recomendada · 🔴 risco bloqueante.

---

2. Pegada de instalação

2.1 DWSIM core

- Aplicação desktop Windows (.NET Framework 4.6.2 / 4.8). Build cross-platform com Eto também disponível para macOS/Linux.
- Caminho de instalação padrão no Windows: `%ProgramFiles%\DWSIM`. O instalador exige admin apenas para gravar em Program Files; tudo o que o DWSIM faz em runtime executa como usuário padrão.
- Dados do usuário: `%AppData%\DWSIM` (configurações, arquivos recentes, plugins).
- Registro: uma única chave `HKCU\SOFTWARE\DWSIM`. Usada para armazenar tokens OAuth do Simulate365 (`AccessTokenV2`, `RefreshTokenV2`) quando o usuário faz login no recurso de nuvem Simulate365; limpa no logout.
- Nenhum serviço Windows, nenhuma tarefa agendada, nenhum mecanismo de auto-update, nenhum driver, nenhum componente de kernel.

2.2 DWSIM Patreon Extensions

- Assemblies adicionais copiados para a pasta de instalação existente do DWSIM. Sem instalador separado, sem pegada própria no Registro.
- Distribuídos apenas a apoiadores com doação mensal recorrente ativa no Patreon. Não fazem parte da build pública do DWSIM.
- Inclui o cliente do AI Convergence Assistant e o runtime TensorFlow.NET para inferência local.

2.3 dwsim-assistant

- `.exe` único PyInstaller. Sem etapa de instalação separada — fica ao lado do binário do DWSIM e é iniciado sob demanda quando o usuário abre o painel do AI Assistant.
- Arquivo de configuração por usuário: `%AppData%\DWSIM\assistant.env` (opcional; não é criado pelo instalador).
- Dados locais: pasta `logs/` e arquivos `*.jsonl` de conversas ao lado do `.exe`. O mirror Supabase é opcional e desligado por padrão.
- Sem serviço, sem auto-update.

---

3. Comportamento de rede

Todas as chamadas de saída estão listadas abaixo. Nada além do que está aqui acontece.

3.1 DWSIM core

Endpoint	Finalidade	Quando dispara	Dados enviados
`dashboard-service.simulate365.com`, `excel-runner-service.simulate365.com`, `sensitivity-study-service.simulate365.com`, `take-home-exams-service.simulate365.com`	Sincronização Simulate365, Excel runner, entrega de provas	Apenas após o usuário fazer login no Simulate365 dentro do DWSIM	Tokens OAuth, arquivos de fluxograma que o usuário enviar explicitamente, perfil básico
`login.microsoftonline.com`	Fluxo OAuth do Azure AD usado pelo Simulate365	Mesmo do anterior	Handshake OAuth padrão
`dwsimlicensingcheck.azurewebsites.net`, `dsw-license-manager.azurewebsites.net`	Validação de licença para recursos Pro	Na inicialização, se houver licenças Pro	Identificador da máquina, chave de licença
`pubchem.ncbi.nlm.nih.gov`	Consulta de metadados de compostos	Manual: quando o usuário adiciona um composto pelo Compound Creator	Nome do composto / CAS / SMILES
`api.crossref.org`	Enriquecimento de metadados de citação	Manual: quando o usuário solicita citações para um relatório	DOI

Se o Simulate365 não for usado e não houver licenças Pro, o DWSIM core não contata nenhum desses endpoints.

3.2 DWSIM Patreon Extensions

Endpoint	Finalidade	Quando dispara	Dados enviados
`dwsim-convergence-assistant-akbsbbd7h3f5enby.brazilsouth-01.azurewebsites.net`	Dados de treinamento de convergência colaborativos	A cada 60 s enquanto uma simulação está rodando, **somente quando** `UploadToServer = true` nas configurações do usuário	JSON comprimido: nomes de componentes, vazões molares, resultados de flash/equilíbrio, e um GUID por máquina para deduplicação. Sem rótulos de fluxograma, sem identidade do usuário, sem metadados de documento

Desabilitar o upload é um único toggle no arquivo de configuração do usuário (`UploadToServer = false`). A TI também pode bloquear o domínio no firewall como defesa em profundidade.

3.3 dwsim-assistant

Endpoint	Finalidade	Quando dispara	Dados enviados
`api.openai.com` , `api.anthropic.com` , `*.cognitiveservices.azure.com` , `openrouter.ai` , endpoint customizado	Completions de LLM para o painel de chat	A cada turno do chat, somente quando o usuário está no backend cloud	Prompt de sistema, histórico da conversa e o XML estrutural do fluxograma ativo (sem imagens)
`dsw-license-manager.azurewebsites.net`	Busca de credenciais por execução	Na inicialização do assistente	Apenas a chave de licença
`*.supabase.co`	Mirror de logging de conversas	Apenas quando o usuário ativa explicitamente o compartilhamento de dados em Settings	Conteúdo das mensagens, argumentos de ferramentas, nome do modelo, UUID da sessão
Servidores MCP do `mcp_servers.json` do usuário	Chamadas de ferramentas externas configuradas pelo usuário	Disparado pelo usuário	O que a ferramenta configurada exigir
Servidor OPC UA (porta 4840)	Browse somente-leitura de tags de processo	Apenas se o usuário configurar um servidor OPC	Browse / read de tags
OSIsoft / AVEVA PI Data Archive	Acesso somente-leitura ao historiador	Apenas se o usuário configurar o PI	Search / read de tags

Os clientes OPC UA e PI nesta build são somente-leitura — fazem browse e read de tags, mas nunca escrevem em sistemas de controle ou historiadores.

---

4. Listeners de entrada

Componente	Endereço de bind	Porta	Autenticação
Servidor Python do dwsim-assistant	`127.0.0.1` (apenas localhost)	5834	Token GUID por execução. Toda chamada HTTP, WebSocket e EventSource precisa do header `X-DWSIM-Token` correspondente (ou parâmetro `?token=` para streams). Requisições sem o token são rejeitadas com HTTP 401
Bridge DWSIM dentro do processo do DWSIM	`localhost`	5002	Mesmo token GUID por execução. Usado pelo assistente para ler o fluxograma ativo e emitir comandos de leitura/escrita para a simulação
DWSIM Automation TCP / Azure Server	varia	varia	Ferramentas independentes — só iniciam se o usuário rodar explicitamente

O token é gerado novo toda vez que o DWSIM inicia o assistente; nunca deixa a máquina local.

---

5. Segredos e credenciais locais

Onde	O que	Sensibilidade	Notas
`HKCU\SOFTWARE\DWSIM` (Registro)	Tokens OAuth de refresh / access do Simulate365	🟡	Presentes apenas após o usuário fazer login no Simulate365. Limpos no logout
Tenant e Client ID Azure AD embarcados no binário	Identificadores OAuth de cliente público usados pelo login Simulate365	🟢	**Não são** segredos — são, por design, identificadores públicos de OAuth public clients
Credenciais Azure / Supabase do dwsim-assistant	Provedor LLM + logging opcional	🟢	Não armazenadas em disco em nenhuma build distribuída. Buscadas em runtime de um endpoint validado por licença e assinado em RSA, mantidas apenas em memória do processo

Não há senhas embarcadas, chaves de API, strings de conexão de banco de dados ou credenciais hardcoded nos binários distribuídos.

---

6. Superfície de execução de código

O que pode rodar código dentro da árvore de processos do DWSIM, e como a TI controla:

Superfície	Quando roda	Padrão	Como desabilitar para uma implantação corporativa
Operações unitárias e pacotes de propriedades nativos	Sempre que um fluxograma é resolvido	Sempre disponível	Não pode ser desabilitado — é a função central do simulador
Runtime IronPython (in-process, usado por scripts anexados pelo usuário a objetos de fluxograma via Script Manager)	Apenas quando o usuário abre um fluxograma que contém scripts anexados, ou anexa um manualmente	Disponível, mas só dispara em scripts anexados pelo usuário	Remover `IronPython.dll`, `IronPython.Modules.dll`, `IronPython.SQLite.dll`, `IronPython.Wpf.dll` e `Microsoft.Scripting.dll` da pasta de instalação
Tool `generic_script` do AI Assistant	Apenas quando o assistente está configurado para usar	**Desabilitada por padrão**. O usuário precisa habilitar pelo painel de Settings do assistente para a sessão atual	Não implantar `dwsim-assistant.exe`, ou distribuir o assistente com scripting travado em build (o binário recusa o toggle nesse caso)
Pasta de plugins	Carregada na inicialização	Vazia por padrão	Restringir a permissão NTFS de escrita em `%AppData%\DWSIM\Plugins`

Execução de Python externo (`python.exe` via Python.NET) e de subprocesso Octave **não** fazem parte desta distribuição. As DLLs antigas `DWSIM.Libraries.PythonLink.dll` e `DWSIM.Libraries.OctaveSharp.dll` foram removidas do build.

Nada na distribuição baixa código de um local remoto, faz auto-update ou carrega DLLs de fora da pasta de instalação.

---

7. Componentes de terceiros

7.1 DWSIM core

UI/runtime: Eto.Forms, GTK# (apenas Linux), OpenTK, SkiaSharp, WebView2 (componente Edge/Chromium), Newtonsoft.Json, SharpZipLib, NetOfficeFw (interop com Excel). Nuvem: Microsoft.Graph e Microsoft.Identity.Client (MSAL) — usadas apenas pelo recurso Simulate365.

7.2 DWSIM Patreon Extensions

Adicionam TensorFlow.NET e o runtime nativo TensorFlow correspondente para inferência local no convergence assistant. LiteDB acompanha apenas o componente server-side (não faz parte de nenhuma instalação cliente).

7.3 dwsim-assistant

Conjunto pinado de pacotes Python (ver `requirements.txt` distribuído com o fonte). Dependências de runtime notáveis: FastAPI + Uvicorn (servidor web), httpx (cliente HTTP), cryptography (verificação de assinatura RSA), pythonnet (bridge .NET, apenas Windows), cliente MCP, asyncua (OPC UA), PIconnect (OSIsoft PI). Todas as dependências estão fixadas em versões exatas.

7.4 Assinatura de código

O `.exe` do DWSIM, as DLLs das Patreon Extensions e o `dwsim-assistant.exe` não são assinados com Authenticode pelo publicador. Ambientes corporativos com AppLocker ou WDAC, portanto, precisam de uma exceção baseada em hash para esses binários ou de um wrapper de assinatura corporativa aplicado no momento da implantação.

---

8. Dados que saem da estação

Canal	O que	Quando
Simulate365	Arquivos de fluxograma completos	Apenas quando o usuário faz upload explicitamente pelo Simulate365
Upload do convergence assistant (Patreon Extensions)	Parâmetros de simulação anonimizados	A cada 60 s durante uma simulação, somente quando `UploadToServer = true`
Provedor LLM (assistente)	XML estrutural do fluxograma ativo e a conversa do chat	A cada turno do chat enquanto o backend cloud estiver selecionado
Mirror Supabase (assistente)	Histórico de conversas e argumentos de ferramentas	Apenas quando o usuário ativa o opt-in pela UI de Settings
OPC UA / PI (assistente)	Requisições de leitura de tags ao servidor configurado pelo usuário	Disparado pelo usuário; somente leitura — sem escrita

O DWSIM não coleta nem transmite dados biométricos, keystrokes, screenshots de outros aplicativos, conteúdo do sistema de arquivos fora de suas próprias pastas, nem PII além do que o usuário digita no login do Simulate365 ou no painel de chat.

---

9. Política de firewall e configuração corporativa recomendada

Uma regra de saída "deny by default" para os processos do DWSIM e do dwsim-assistant, combinada com a allowlist abaixo, é suficiente para confinar a instalação aos casos de uso que seus usuários realmente precisam.

```
# Sempre necessário se o usuário tiver licença paga
dwsimlicensingcheck.azurewebsites.net
dsw-license-manager.azurewebsites.net

# Necessário apenas se o Simulate365 for usado
*.simulate365.com
login.microsoftonline.com
graph.microsoft.com

# Opcional — consultas químicas manuais
pubchem.ncbi.nlm.nih.gov
api.crossref.org

# Necessário apenas se o AI Assistant for usado (escolher um provedor)
api.openai.com                  # OU
api.anthropic.com               # OU
<resource>.cognitiveservices.azure.com

# Bloqueio recomendado (sem motivo de negócio em estação corporativa)
dwsim-convergence-assistant-akbsbbd7h3f5enby.brazilsouth-01.azurewebsites.net
*.supabase.co
```

Políticas adicionais que um administrador pode aplicar:

- Definir `UploadToServer = false` no arquivo de configuração das DWSIM Patreon Extensions sob `%AppData%\DWSIM` e proteger o arquivo com ACLs de somente-leitura ou uma preferência de Group Policy.
- Monitorar `HKCU\SOFTWARE\DWSIM` em busca de tokens OAuth em estações não aprovadas para Simulate365.
- Em ambientes onde a assistência por IA não é desejada, não implantar `dwsim-assistant.exe` (o DWSIM funciona normalmente sem ele) ou remover `DWSIM.Extensions.AI.Assistant.dll`.
- Em ambientes onde scripting in-app (IronPython anexado a objetos de fluxograma) não é desejado, remover `IronPython.dll`, `IronPython.Modules.dll`, `IronPython.SQLite.dll`, `IronPython.Wpf.dll` e `Microsoft.Scripting.dll` da pasta de instalação. Execução de Python externo e subprocesso Octave já não fazem parte da distribuição.
- Aplicar um wrapper corporativo de assinatura Authenticode (ou exceção AppLocker baseada em hash) aos binários do DWSIM e do dwsim-assistant.