Engineer

Gemini

ハイパーテキストを配信する軽量インターネットプロトコル

どのようなものか

Geminiは文書を取得するためのアプリケーションプロトコルです。やり取り全体は、一つの暗号化接続を開き、クライアントがアドレスを一つ送り、応答を一つ受け取り、接続を閉じるだけです。公式文書はGeminiプロトコルと呼び、アドレスは `gemini://` で始まります。このプロトコルで配信するサイトをカプセル、そのネットワークをGeminispaceと呼びます。

プロジェクトは2019年6月、創設者Solderpunkの下で始まりました。現在も「善意の独裁者」を務めています。仕様のバージョン0.24.1はCC0によってパブリックドメインへ提供され、プロトコルを「機能を削ったHTTPではなく、Gopher [RFC1436]の段階的な改良」と説明します。公式FAQは逆の側から、「HTTP 0.9とHTTP 1.0の中間のようなもの」と比較します。リクエストは一種類だけで、おおよそ `GET` に相当します。

クライアントはTCPポート1965に接続し、TLSハンドシェイクを完了します。この番号は、1965年3月に実施されたGemini計画初の有人飛行にちなみます。双方ともTLS 1.2以降に対応しなければなりません。クライアントは絶対URLに復帰文字と改行文字を付けて送り、URLの上限は1,024バイトです。サーバーはヘッダーを一行で返します。二桁のステータスコード、空白、コードごとに意味が変わる短いテキストです。コードは六つのグループに分かれ、10–19は入力要求、20–29は成功、30–39はリダイレクト、40–49は一時的な失敗、50–59は恒久的な失敗、60–69はクライアント証明書を扱います。成功時のテキストはMIMEタイプで、その後に本文が生のバイト列として続きます。圧縮もチャンク分割もありません。最後にサーバーがTLSのclose_notifyメッセージで接続を閉じるため、クライアントは応答をすべて受信したと判断できます。

どんな種類のファイルも運べますが、専用の文書形式gemtextと一緒に設計されました。その形式は `text/gemini` で配信します。

概念を判断に生かす

「Geminiとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 表示が速く、検索で見つけやすく、文章を変えるたびに料金が発生せず、自社の担当者が編集できるサイトをつくります。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 HTML/HTML5でWebアプリケーション100個を開発 — 100 構築したWebアプリケーション

HTML/HTML5でWebアプリケーション100個を開発 お問い合わせ →

なぜ重要なのか

現実の判断は、サイトをGeminiへ移すかどうかではないことがほとんどです。FAQは、ウェブやGopherの置き換えを目的としないと述べ、同じ内容を二つ、三つの方式で公開することをbihosting、trihostingと呼びます。運営者にとっての問いは、二つ目の入口を開けておく価値があるかです。

利点は設計に組み込まれたプライバシーです。FAQは、Geminiには「Cookie、Referer、User-Agentヘッダーに相当するものがなく、今後も決して持たない」と述べます。リクエストはURLだけなので、運営者が記録したくてもサーバーが記録できるものはごくわずかです。読者のクライアントにも、情報を送り返すスクリプトはありません。FAQは設計上の優先事項に「防御のための非拡張性」を挙げます。リクエストと応答の形式には新しいヘッダー欄を加える余地がなく、将来の版にも追跡を持ち込む場所がありません。プレーンテキストを公開する人には、これが魅力のすべてです。

代償は、HTTP 1.0以降にウェブが獲得した機能の多くを手放すことです。Cookieがないため、再訪者を識別するカプセルは代わりにクライアント証明書を求め、読者が作成して保管します。既定では認証局を使いません。仕様はTrust on First Use、TOFU、初回利用時の信頼を強く推奨します。初回にサーバーが示した証明書は、どんなものでも受け入れ、その後の接続の基準として固定する方式です。サーバー証明書を、長期にわたる約束として扱うことになります。暗号化でページは隠れますが、サーバー名はTLSのServer Name Indicationで伝わり、FAQもホスト名が「現在もなお漏れる」と認めます。ブラウザーではなく専用クライアントが必要で、読者層は小さくなります。応答にサイズもチェックサムもないため、大きいファイルにも向きません。

対話的なフォーム、パスワード付きアカウント、アクセス解析、豊かなレイアウトが必要なチームにはウェブが適しています。文書を公開し、Cookieやクライアント側スクリプトのない経路を求めるなら、内容がすでにファイルとして存在する限り、少ない費用でGeminiを追加できます。

実際の活用例

engineer.company自体も、`gemini://engineer.company`にあるカプセルです。クレジットは、Geminiに応答するAgate、ウェブのCaddy、GopherのGophernicus、onionサービスのTorを挙げます。四つとも同じフォルダーを読むため、一回のデプロイですべてを更新し、互いの内容がずれることはありません。FAQのbihostingを四経路へ広げ、最小限のコストで実現した形です。カプセルを加えても、デプロイのたびにファイルが書き込まれる既存のフォルダーを読むプログラムが一つ増えるだけです。別のサイトを用意して内容を合わせ続ける必要はありません。

同じページは、先の制約も率直に説明します。GeminiとGopherにはそれぞれ専用クライアントが必要で、Gopherでは読んだページが平文で伝わります。Geminiはそこを暗号化しますが、利用中のネットワークには、どのサーバーへアクセスしたかが見えます。FAQのサーバー名の漏えいを、設計者ではなく読者の側から説明したものです。

プライバシーは、アプリケーションのアクセスログとOSのログを区別しています。GeminiとGopherのアプリケーションはアクセスログを保持せず、2026年9月以降、日々のカウンターには要求の合計数だけを記録しています。Gopherには接続元アドレスを含むOSの接続ログもあり、その保存期間は30日です。

問題が起きるところ

Geminiの問題の多くは、応答の最初の一行に現れます。OpenSSLだけで手作業でも簡単に読めます。

```
printf 'gemini://engineer.company/\r\n' |
  openssl s_client -quiet -connect engineer.company:1965 \
    -servername engineer.company
```

返る最初の一行がヘッダーで、このカプセルでは`20 text/gemini;lang=en`です。その後にページが続きます。

再訪した読者に、カプセルが攻撃されている可能性を示す警告が出ます。 仕様に従うクライアントは、最初に受け取った証明書のフィンガープリントと有効期限を保存します。その期限より前に別の証明書が現れると、「中間者攻撃の可能性を示す証拠」と見なします。デプロイごとに新しい証明書を作ったり、カプセルを鍵なしに新しいマシンへ移したりすると、すべての常連読者へ警告が出ま

す。鍵と証明書もサーバーのほかの内容とともにバックアップし、有効期間の長い証明書を発行し、クライアントが記録済みの期限を過ぎてから交換してください。

検索欄がステータス59で応答します。 ステータス10の応答は一行の入力を求め、クライアントは同じURLのクエリー部分へ、パーセント符号化して送り返します。URLの上限は1,024バイトで、サーバーは超過を不正なリクエストとして拒否しなければなりません。長い回答はすぐ上限に達し、ASCII以外の文字はさらに早く達します。キリル文字一字はUTF-8で二バイト、符号化後は六バイトです。回答を短く求め、パスフレーズなどの機密入力にはステータス11を使い、クライアントが画面へ表示しないようにしてください。

ページを移るとログアウト状態になります。 Cookieがないため、ウェブと同じ方法で保つセッション状態は、次のリクエストまで続きません。相手を識別するカプセルはステータス60を返し、読者のクライアントが証明書を作り、後のリクエストでサーバーが認識します。仕様は証明書の範囲を、要求したホストとポート、要求したパスとその配下全体に限ります。識別が必要なページは同じパスの下へ置いてください。読者の同意が必要なので、証明書の目的もページで説明してください。

大きいダウンロードが途中で切れても、誰も気付きません。 応答には長さもチェックサムもなく、途中で止まっても、close_notifyがないことを除けば、完了した場合に似て見えます。実装のベストプラクティスは、100MiBを超えるファイルを別の方法で配信するよう勧めます。大きいファイルはHTTPSへ置き、カプセルからリンクしてください。Geminiページはどんなスキームにもリンクできます。

参考資料

- Geminiネットワークプロトコル仕様 — リクエスト、全ステータスコード、TLS、証明書の固定をまとめた短い文書です。
- Project GeminiのFAQ — 目的、HTTPとGopherとの比較、TOFUは弱いという批判に対する、プロジェクト自身の回答です。
- Gemini実装のベストプラクティス — ファイル名、大きいファイル、リダイレクト制限、TLS設定です。
- Project Geminiの歴史 — Gopherspaceでの2019年の発想が独自のネットワークになった経緯を、プロジェクト自身が説明します。

用語集

実際の制約に合わせて選ぶ

業務の流れ、既存のインターフェース、変更を必要とする利用者をお知らせください。検証可能な納品内容と、お客様のチームが保守を続けられる引き継ぎ内容を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「Geminiとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/gemini/