sgClaw 浏览器对接标准（Chromium ↔ sgClaw）

> 适用范围：P1a（Rust）与 P2（Chromium C++）联调开发。 目标：双方只要严格按本文档实现，即可稳定联调。

1. 协议边界与责任

- 单一事实来源：`docs/L2-核心模块与接口契约层.md` 第 5 章（5.1~5.4）。
- 协议版本冻结：`1.0`；字段、action、错误码变更均视为协议变更。
- P1a 负责：`seq` 生成、command 组包、HMAC 计算、response 关联。
- P2 负责：message 解析、Schema 校验、MAC 检查、CommandRouter 执行、结构化回包。

2. Wire Contract（双方 MUST）

项目	强约束	违规错误码
传输层	STDIO + JSON Line（每行一条完整 JSON）	`PIPE_INVALID_JSON`
编码	UTF-8	`PIPE_INVALID_JSON`
消息大小	单条消息 `<= 1MB`	`PIPE_MESSAGE_TOO_LARGE`
序列号	`seq` 从 1 开始、严格递增、不可重复	`PIPE_SEQ_DUPLICATE` / `PIPE_SEQ_OUT_OF_ORDER`
安全字段	command 必含 `security.expected_domain` 与 `security.hmac`	`PIPE_HMAC_INVALID` / `MAC_*`
一问一答	一个 `seq` 必须且只能对应一个 response	`INTERNAL_UNKNOWN`

3. 握手协议（启动门禁）

1. Browser 启动 sgClaw 子进程。
2. Browser 发送：`{"type":"init","version":"1.0","hmac_seed":"<hex>"}`。
3. sgClaw 返回：`{"type":"init_ack","version":"1.0","agent_id":"<uuid-v4>","supported_actions":[...]}`。
4. Browser 超时 `5000ms` 未收到 `init_ack`：Kill 子进程并置状态 `Crashed`。
5. 任一方 `version` 不一致：立即失败，不进入 Running。

4. 命令/响应字段标准

- command 必填：`seq`、`type=command`、`action`、`params`、`security`。
- response 必填：`seq`、`type=response`、`success`。
- 失败 response 必填：`error.code`、`error.message`（禁止纯文本错误）。
- `action` 与 `params` 必须通过 L2 的枚举和 Schema 校验。

标准 command 示例：

```
{"seq":12,"type":"command","action":"click","params":
{"selector":"#submit"},"security":
{"expected_domain":"erp.example.com","hmac":"<hex>"}}
```

5. HMAC 统一规则（避免两端实现不一致）

- 算法：`HMAC-SHA256`，输出小写 hex。
- 密钥：由 `hmac_seed` 派生后在会话内固定。
- 签名原文（canonical string）：

```
<seq>\n<action>\n<stable_json(params)>\n<expected_domain>
```

- `stable_json(params)`：键名按字典序、无多余空格、UTF-8 编码。

6. 错误处理与重试矩阵

错误类型	重试策略
`PIPE_*`	不重试，直接失败
`MAC_*`	不重试，等待配置/人工确认
`CMD_SELECTOR_TIMEOUT`	最多重试 2 次（500ms、1000ms）
`CMD_NAVIGATION_FAILED`	最多重试 1 次（1000ms）
`INTERNAL_*`	最多重试 1 次，仍失败则熔断

- 同一 action 连续失败 `>10` 次：触发熔断并通知 UI。

7. 联调验收（全部通过才算完成）

- [ ] `init -> init_ack` 连续 100 次成功率 100%。
- [ ] 版本不匹配时稳定失败并返回可读日志。
- [ ] `seq` 重复/乱序场景可复现并返回标准错误码。
- [ ] >1MB 消息可稳定被拒绝。
- [ ] 核心 action（click/type/navigate/getText）成功率 `>=99%`。
- [ ] 所有失败场景均返回结构化 `error.code` + `error.message`。
- [ ] 日志可按 `seq` 贯通请求、执行、响应。