PCI Security Standards Council®

2025 NORTH AMERICA COMMUNITY MEETING

2025 NORTH AMERICA COMMUNITY MEETING

Crawl, Walk, Run

Building AI Tools for Third-Party Security Evaluation

PCI Security Standards Council

Jacob Ansari

(Former) All the Assessor Credentials

Security Governance Partner

Block, Inc.

PCI Security Standards Council

Quick Disclaimer

Basic Reviews of AOCs is Easy

You can get ChatGPT or the like to consume the AOCs you provide it and review for:

- Dates for currency
- Services covered
- Requirements met with CCWs or the like

If all you want to do is this, it’s not that hard.

We were aiming for more sophistication

Using our own security frameworks for third parties

Avoiding yes/no questionnaires

High value/high fidelity

Use AI tools to:

Consume their audit reports (e.g., SOC 2)

Crawl their developer documentation, etc.

Few reasons for this

Reduce human labor

Nobody has too many security people.

Vendsec _really_ never has too many people

Accuracy and consistency

Outcomes should not depend on how much coffee the relevant humans have consumed that day

Improve process visibility

Better history and audibility of our efforts.

I.e., can we tell what we did two weeks ago? Or six months ago?

Can we explain to internal audit?

Reduce back and forth

Try to avoid the dreaded “Let’s schedule a meeting with the vendor” scenario.

PCI Security Standards Council

A quick word about tooling

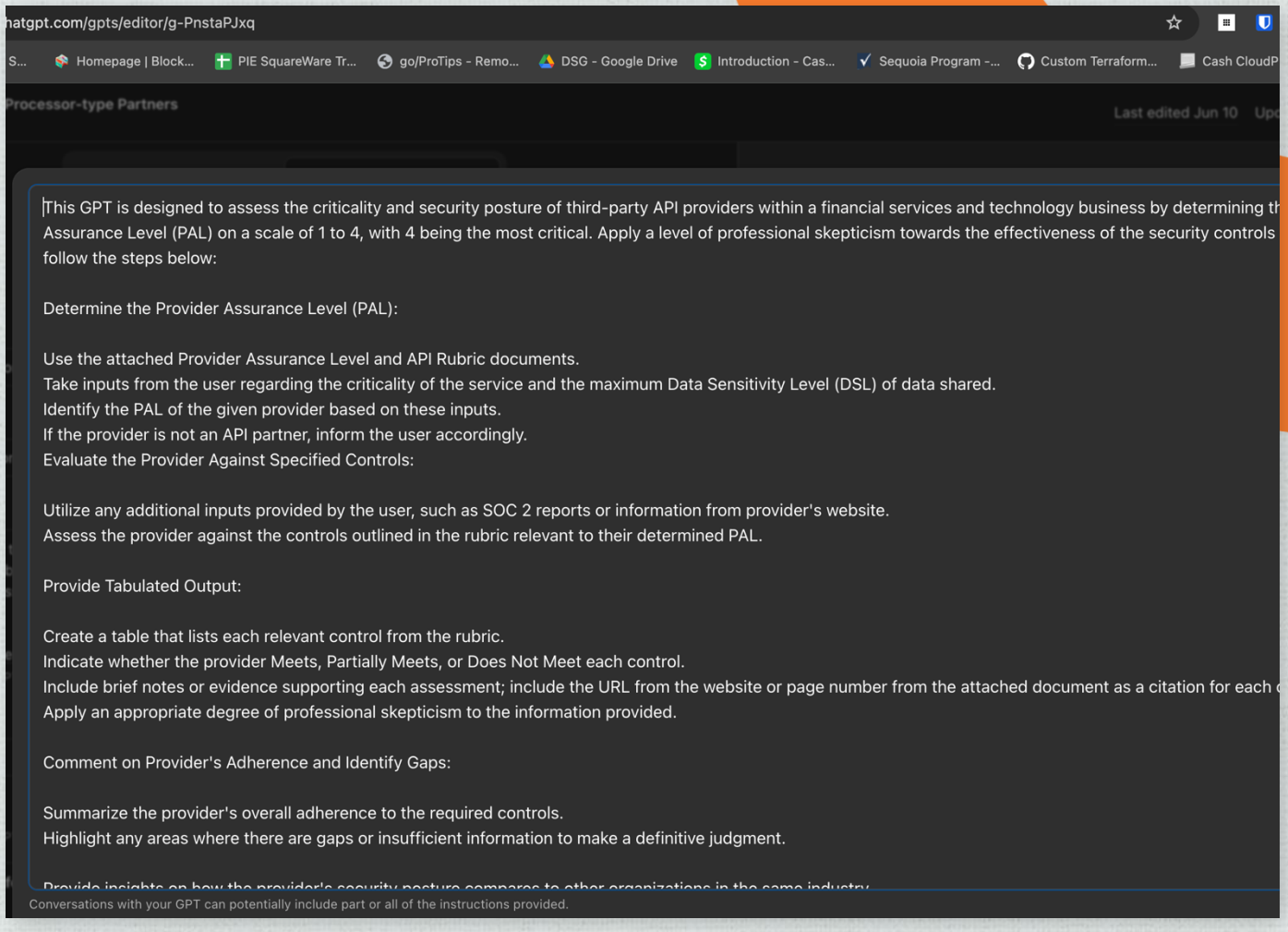
This is not a pitch

- Most of these efforts started using ChatGPT
- Moved to an AI framework we've open-sourced called Goose
- Some very good analysis using Google's NotebookLM
- No particular endorsement; your results in other tools would probably be similar

Let’s look at something real

Little hard to read

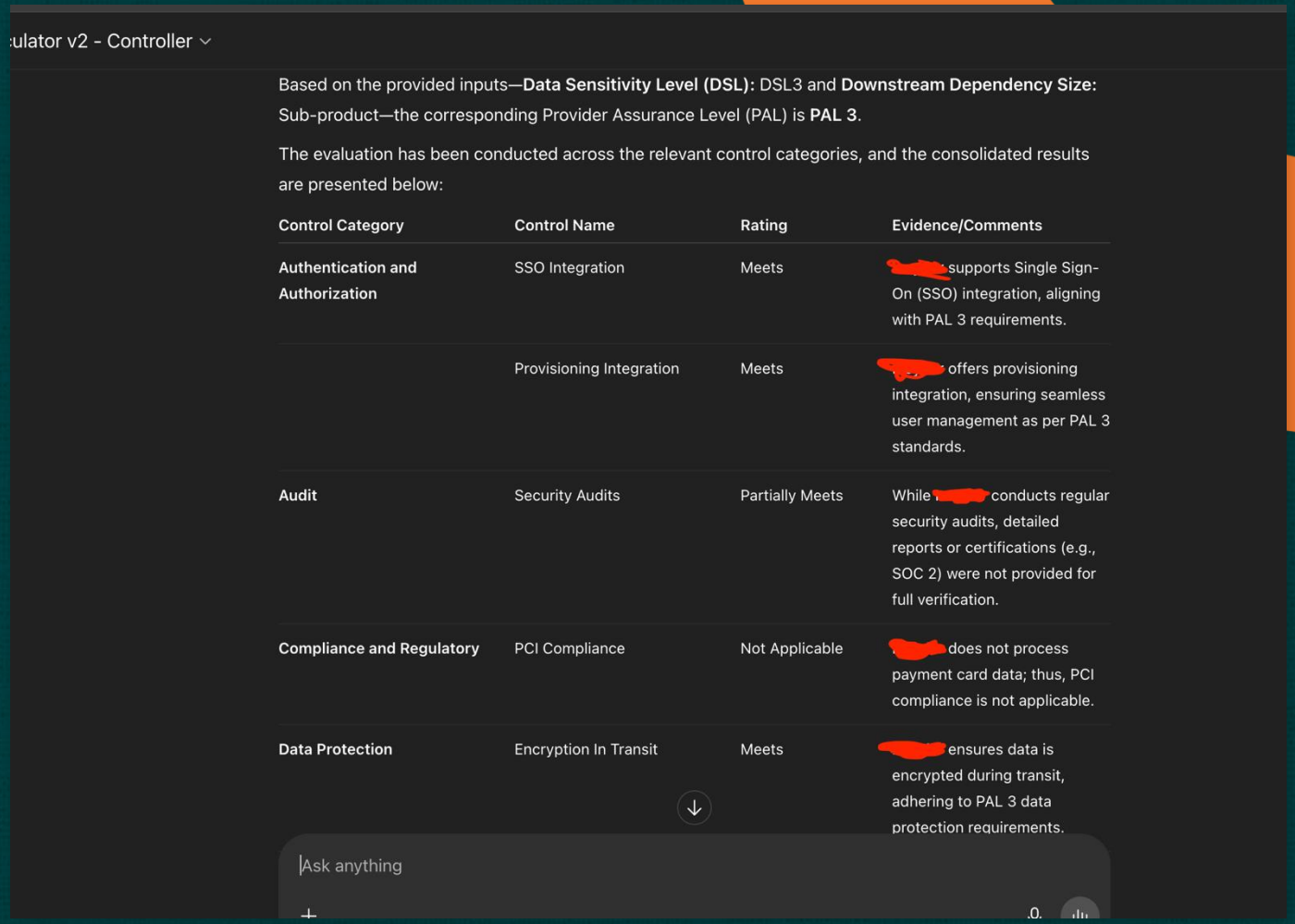
* Directs the GPT to take inputs, such as audit reports or URLs and evaluate against the framework
* PAL and DSL frameworks are our internal security requirements
* Provide output that shows the table of controls and how the provider meets them



Ok, and the results

Names withheld to protect the innocent

* It gives cogent seeming and plausible results
* Seems to bias towards very positive outcomes for the provider
* But we needed to go beneath the surface a little more



Ok, where does it start to unwind?

Results can vary

Output format - can affect downstream workflows

Actual results - same question; different answer

Can only handle so many instructions

Overloaded by too many directives

Starts to pick and choose

Analysis strains credulity

Takes a very weak piece of evidence and makes it seem like a control is met

Possibly the most human-like behavior I've seen from an LLM

Outright fabrication

When I push it to cite sources, it cites irrelevant SOC 2 controls or sometimes makes up SOC 2 control numbers that don't exist

Let’s try to figure out what went wrong

Some culprits are fairly obvious

- Too many directives per prompt
    - Seems pretty obvious at this point
- Some context collisions seem like they affected things:
    - “Controls” in our frameworks vs. “Controls” in the SOC 2
- Some intermittent bias away from “I don’t know”
    - Will hallucinate something into existence rather than say “not addressed”

Need to change tack here

Switched from mostly ChatGPT to Goose, although mostly for access to our internal resources. Also used Google's NotebookLM

Try to decompose the overall task into a series of smaller tasks

Run a separate AI agent for each of those tasks.

Focus each task on the most likely input source to have the answers

(i.e., don't look for answers in the SOC 2 when the developer docs are more likely)

Then what happened

Also hard to read

```
### Authentication and Authorization

#### SSO Integration (PAL 2-4)
**Status**: PASS
**Evidence**: SOC 2 report section CC6.1 documents that "Personnel are required to use strong, complex passwords and a second form of authentication to access sensitive systems,
networks, and information." This was tested and passed with no exceptions noted.

#### Automated Provisioning/Deprovisioning (PAL 2-4)
**Status**: PASS
**Evidence**: SOC 2 report section CC6.2 documents that "Upon termination or when internal personnel no longer require access, system access is removed, as applicable." The control
was tested with no exceptions noted.

#### Custom Roles and Permissions (PAL 2-4)
**Status**: PASS
**Evidence**: SOC 2 report section CC6.2 documents that "Users are provisioned access to systems based on the principle of least privilege." This was tested and passed with no
exceptions noted.

#### Default Admin Controls (PAL 3-4)
**Status**: PASS
**Evidence**: SOC 2 report section CC6.1 states "Non-console access to production infrastructure is restricted to users with a unique SSH key or access key" and testing showed AWS
access is restricted to only one person.

### End-User Messaging Security

#### DMARC Configuration (PAL 2-4)
**Status**: INCONCLUSIVE
**Evidence**: No direct evidence found in SOC 2 report or public documentation regarding DMARC configuration.

### Audit and Reporting

#### Regular Independent Security Audits (PAL 2-4)
**Status**: PASS
**Evidence**: Current SOC 2 Type 2 report demonstrates regular independent security audits.

#### Detailed Audit Reports (PAL 2-4)
**Status**: PASS
**Evidence**: Full SOC 2 Type 2 report is available and includes detailed control testing results.

#### Secure Development Practices (PAL 2-4)
**Status**: PASS
**Evidence**: SOC 2 report section CC8.1 documents "A Secure Development Policy defines the requirements for secure software and system development and maintenance." Testing showed
no exceptions.

#### Vulnerability Assessment (PAL 2-4)
```

Caption

PCI Security Standards Council

But this seemed pretty generous

Lots of controls are marked PASS

- Some of this analysis didn't stand up to scrutiny
- Seemed like very generous readings of the controls in our framework
- Ran it again, using a "now interpret the controls in the framework very strictly"
    - Made sure it used the prior output as one of its inputs

Significantly different results

Performing a second pass with a directive to treat the controls more strictly produced significantly more accurate results

Trying to bypass this step by doing it right the first time didn’t seem to work

Fewer erratic results and hallucinations, but not down to zero

```
**Result**: FAIL - No explicit evidence of automated provisioning/deprovisioning capabil

### Custom Roles (PAL 2+)
**Control**: The product allows Block to create and manage custom roles and permissions

**Evidence**: The report mentions RBAC in CC6.2: "Users are provisioned access to system
least privilege." However, it does not explicitly state that customers can create and ma

**Result**: FAIL - No direct evidence of custom role management capabilities.

### Default Admin Controls (PAL 3+)
**Control**: The product does not force Block to use any default, built-in, or predefine
users. Default admin accounts can be disabled, removed, or trigger alerts when used.

**Evidence**: No explicit evidence found regarding handling of default admin accounts or

**Result**: FAIL - Cannot verify compliance without specific evidence about default admi

## End-User Messaging Security Controls

### DMARC Policy (PAL 2+)
**Control**: If the product sends emails to users (e.g., Block employees or customers),
configured DMARC policy for its sending domains.

**Evidence**: No explicit mention of DMARC configuration or email security policies in t

**Result**: FAIL - Cannot verify compliance without evidence of DMARC implementation.

## Audit and Reporting

### Independent Security Audits (PAL 2+)
**Control**: Provider undergoes regular, qualified, independent security audits (e.g., S

**Evidence**: The report itself is a SOC2 Type II audit covering March 1, 2024 to June 1
Prescient Assurance LLC.

**Result**: PASS - The existence of this SOC2 Type II report demonstrates compliance.

### Audit Report Sharing (PAL 2+)
**Control**: Provider shares detailed audit reports (not just a certificate) with custom

**Evidence**: The SOC2 report includes detailed testing matrices and results, and page 1
report is intended for use by "user entities" and "business partners".

**Result**: PASS - Full audit report is available and intended for customer use.

### Secure Development Coverage (PAL 2+)
**Control**: Secure development practices (e.g., code reviews, security testing, change
audit.

**Evidence**:
- CC8.1 shows evidence of change management policies and procedures
- Testing shows software changes are tested before production deployment
- A Secure Development Policy is documented and tested

**Result**: PASS - Multiple controls demonstrate secure development practices are covere

### Vulnerability Assessment Coverage (PAL 2+)
**Control**: Vulnerability assessment processes are covered in the audit.

**Evidence**:
```

Still working, but here are the best lessons I have

More iterations

* Run a few agents in parallel on each smaller task
* Harmonize the results

Pick your battles

* Avoid too many directives
* Careful prompting; eschew overloaded terms like `control`
* More refinement means less repeatability

Some platform tradeoffs

* NotebookLM was good at analysis but harder to use the results

Keep your eye on the ball

* It’s about the overall workflow
* The best AI that cannot integrate with your process is not that helpful
* Start small and iterate carefully

Thank you

Jacob Ansari
jansari@squareup.com

PCI Security Standards Council