

CPF Field Kit - English (en-US)

Contents

Overview	1
Language Information	1
Directory Structure	2
JSON Structure	2
Usage	3
Loading in Interactive Client	3
Validation	3
Master Reference: Indicator 1.3	4
Creating New Indicators	4
Translation Workflow	4
Categories Reference	5
Quality Standards	5
Support	5
Version Control	6

Overview

This directory contains the **master reference** English language versions of all CPF (Cybersecurity Psychology Framework) indicator field kits. These JSON files serve as the authoritative source for all translations and implementations.

Language Information

- **Language Code:** en-US
- **Language:** English (United States)
- **Status:** Master/Source Language

- **Total Indicators:** 100 (10 categories × 10 indicators each)

Directory Structure

```
en-US/
  1.x-authority/          # Authority-Based Vulnerabilities
    indicator_1.1.json
    indicator_1.2.json
    indicator_1.3.json # Master reference example
    ...
  2.x-temporal/           # Temporal Vulnerabilities
  3.x-social/            # Social Influence Vulnerabilities
  4.x-affective/          # Affective Vulnerabilities
  5.x-cognitive/          # Cognitive Overload Vulnerabilities
  6.x-group/              # Group Dynamic Vulnerabilities
  7.x-stress/             # Stress Response Vulnerabilities
  8.x-unconscious/        # Unconscious Process Vulnerabilities
  9.x-ai/                 # AI-Specific Bias Vulnerabilities
  10.x-convergent/        # Critical Convergent States
```

JSON Structure

Each indicator JSON file follows this standardized schema:

```
{
  "indicator": "X.Y",
  "title": "INDICATOR X.Y FIELD KIT",
  "subtitle": "Specific Vulnerability Name",
  "category": "Category Name",
  "version": "1.0",
  "description": { ... },
  "scoring": {
    "method": "bayesian_weighted",
    "weights": { ... },
    "question_weights": { ... }
  },
  "detection_formula": { ... },
  "data_sources": [ ... ],
  "interdependencies": { ... },
  "sections": [
    {
      "id": "quick-assessment",
      "title": "Quick Assessment",
      "items": [ ... ]
    },
    {
      "id": "client-conversation",
```

```

    "title": "Client Conversation Guide",
    "subsections": [ ... ]
  }
],
"validation": { ... },
"remediation": { ... },
"risk_scenarios": [ ... ],
"metadata": {
  "language": "en-US",
  "language_name": "English (United States)",
  "is_translation": false,
  "translated_from": null,
  "translation_date": null,
  "translator": null,
  "created_date": "YYYY-MM-DD",
  "last_modified": "YYYY-MM-DD",
  "version_history": []
}
}

```

Usage

Loading in Interactive Client

Use the CPF Field Kit interactive client to load and assess indicators:

1. Open `cpf_client_json.html` in a web browser
2. Click “Load JSON” button
3. Enter indicator code:
 - X.Y → Loads from en-US (default)
 - X.Y-EN → Explicitly loads from en-US
 - X.Y-IT → Loads Italian version

Examples: - 1.3 → Loads indicator 1.3 from en-US (default) - 1.3-EN → Explicitly loads indicator 1.3 from en-US - 2.5 → Loads indicator 2.5 from en-US

Validation

To validate any JSON file, use the “Validate JSON” button in the client after loading, or:

```

// In browser console or Node.js
const { validateCPFJSON } = require('./validator.js');
const data = require('./en-US/1.x-authority/indicator_1.3.json');
const result = validateCPFJSON(data);
console.log(result);

```

Master Reference: Indicator 1.3

Indicator 1.3 (Authority Figure Impersonation Susceptibility) serves as the **master reference** for:

- Complete schema structure
- Question formatting
- Scoring methodology
- Detection formula patterns
- Interdependency modeling
- Metadata standards

All other indicators should follow the patterns established in indicator 1.3.

Creating New Indicators

When generating new indicator JSONs:

1. Use **PROMPT_TEMPLATE.md** from the parent directory
2. Reference **indicator 1.3** as the schema example
3. **Validate** using `validator.js` before committing
4. Follow **naming convention**: `indicator_X.Y.json`
5. Set **metadata correctly**:

```
"metadata": {  
  "language": "en-US",  
  "language_name": "English (United States)",  
  "is_translation": false,  
  "translated_from": null,  
  ...  
}
```

Translation Workflow

To create translations from these master files:

1. Copy the en-US JSON to the target language directory
2. Translate all **user-facing text** (questions, descriptions, guidance)
3. **Keep in English**: JSON keys, formulas, research citations
4. Update metadata:

```
"metadata": {  
  "language": "it-IT",  
  "language_name": "Italian (Italy)",  
  "is_translation": true,  
  "translated_from": "en-US",
```

```

"translation_date": "YYYY-MM-DD",
"translator": "Name"
}

```

5. Validate using `validator.js`
6. Test in interactive client with X.Y-IT format

Categories Reference

Code	Category	Description
1.x	Authority-Based Vulnerabilities	Obedience to authority, hierarchical structures
2.x	Temporal Vulnerabilities	Time pressure, urgency-induced risks
3.x	Social Influence Vulnerabilities	Cialdini's principles, peer pressure
4.x	Affective Vulnerabilities	Emotional and attachment-based risks
5.x	Cognitive Overload Vulnerabilities	Information overload, decision fatigue
6.x	Group Dynamic Vulnerabilities	Groupthink, collective unconscious
7.x	Stress Response Vulnerabilities	Fight/flight/freeze/fawn responses
8.x	Unconscious Process Vulnerabilities	Shadow projection, defense mechanisms
9.x	AI-Specific Bias Vulnerabilities	Human-AI interaction risks
10.x	Critical Convergent States	Perfect storms, cascade failures

Quality Standards

All JSON files in this directory must:

- Pass `validator.js` with 0 errors
- Follow indicator 1.3 schema pattern
- Include complete metadata
- Have properly formatted questions (8-12 for quick assessment)
- Include validated detection formulas
- Contain Bayesian interdependencies where applicable
- Reference peer-reviewed research
- Provide practical remediation solutions

Support

- **Documentation:** `/auditor field kit/interactive/PROMPT_TEMPLATE.md`
- **Validator:** `/auditor field kit/interactive/validator.js`
- **Client:** `/auditor field kit/interactive/cpf_client_json.html`
- **Repository:** <https://github.com/xbeat/CPF>

Version Control

This is the **master branch** for English language content. All updates should:

1. Be validated before commit
2. Update version numbers appropriately
3. Document changes in metadata.version_history
4. Maintain backward compatibility where possible

Last Updated: 2025-11-08 **Framework Version:** CPF v1.0 **Status:** Active Development