



RULE-BASED CROSS-MATCH

HOW PRIVATE COMPLIANCE AND COMPATIBILITY CHECKING WORKS

THE SCENARIO

A hospital (MedCare) is about to prescribe a new set of medications to a patient. A pharmaceutical company (PharmaCo) maintains a database of known drug interactions. MedCare won't share its patient's medication list. PharmaCo won't share its proprietary interaction database in raw form. But both need to know: are there any dangerous interactions?

THE PROBLEM

Drug interaction checks typically require one side to see the other's data. Either the hospital sends the patient's medications to a third-party checker (exposing PII), or the pharma company shares its full interaction rules (exposing proprietary research). Neither is acceptable.

HOW IT WORKS

STEP 1: AGREE ON THE RULE LIST

PharmaCo publishes a rule list as a simple CSV. Each row maps one item to another:

```
aspirin, warfarin
metformin, contrast dye
lisinopril, potassium supplements
ibuprofen, lithium
...
```

This rule list is public shared knowledge. It says "if someone has the left item AND someone has the right item, that's a match." It doesn't reveal who has what.

STEP 2: EACH SIDE ENCODES THEIR PRIVATE DATA

MedCare takes the patient's current medications and turns them into an encrypted binary indicator. For each medication the patient is on, the corresponding slot is set to 1.

PharmaCo takes the substances it wants to check against and does the same.

Neither side sees the other's list. Both encrypt locally using the JuLenny toolkit.



STEP 3: UPLOAD AND COMPUTE

Both sides upload their encrypted indicators plus the shared rule list to JuLenny. The platform evaluates each rule pair: for every row in the CSV, it checks whether the left item is in one party's set AND the right item is in the other's. All checks happen entirely on encrypted data.

STEP 4: GET THE RESULT

Count variant: the result is a single number. "3 interactions found." Both sides know there are problems, but not which specific ones (unless they only flagged a few items).

Itemized variant: the result shows exactly which rule pairs triggered. "Row 1 (aspirin + warfarin): match. Row 4 (ibuprofen + lithium): match." The viewing party can see precisely which rules fired.

WHAT NOBODY LEARNS

- MedCare never sees PharmaCo's full interaction database beyond the shared rule list
- PharmaCo never sees the patient's full medication list
- Medications and substances that did NOT trigger any rule are completely hidden
- JuLenny never sees either side's private data

USE CASES BEYOND PHARMA

The same function works for any "does my X interact with your Y under public rules R?" question:

- **Food safety:** A restaurant's ingredient list vs. a customer's dietary restrictions, checked against an allergen rule table. "2 allergen violations found" without the restaurant seeing the full allergy profile or the customer seeing the full recipe.
- **Regulatory compliance:** A company's planned business activities vs. applicable regulatory clauses. "4 compliance issues found" without exposing the full activity plan or the regulator's enforcement priorities.
- **Supply chain:** A manufacturer's bill of materials vs. a supplier's proposed substitutions, checked against a compatibility rule table. "1 incompatible swap found."
- **HR / Talent:** A company's open role requirements vs. a candidate's skills, checked against a role-fit matrix. "5 role matches found."

TWO VARIANTS

Count: returns a single integer. How many rules matched. Good for a quick go/no-go check.

Itemized: returns a per-rule result vector. Exactly which rules triggered. Good for detailed analysis and remediation.



TECHNICAL DETAILS

- **Scheme:** CKKS (approximate arithmetic, used for the rotation-based computation)
- **Multiplicative depth:** 1
- **Rotation keys:** required (derived from the rule list at keysetup time)
- **Max rules:** depends on the number of unique items in the rule list (up to 8,192 slots per ciphertext)
- **Rule list:** plaintext CSV, shared between both parties