

...on Threat Modeling

A Threat Does Not Exist Merely Because It Is in the Catalogue

Alexey A. Nekludoff

AstraVerge Research

E-mail: an@astraverge.org
ORCID: 0009-0002-7724-5762

September 19, 2026

Contents

1	A Catalogue Is Not a Model	1
2	Architecture Precedes Threats	1
3	The DoS Inside the Control Network	2
4	Impossible Is Not Low Risk	2
5	Threat Models as Reachability Models	3

1 A Catalogue Is Not a Model

Threat modeling often begins with a seemingly reasonable question: what threats should be considered for a given system?

A convenient answer is to consult a threat catalogue, select the threats that appear relevant, and then estimate their probability, severity, or risk. This procedure is useful as a checklist, but it contains a subtle methodological inversion.

A threat does not become a property of a system merely because the corresponding attack technique exists.

Consider a denial-of-service attack against a communication channel inside an industrial control system. Denial-of-service attacks certainly exist. Communication channels certainly exist. It therefore appears reasonable to include “DoS attack against ICS communication channels” in a threat model.

But before estimating the severity or probability of such an attack, a more fundamental question must be answered:

From what reachable state of the system can an attacker perform it?

If an attacker cannot reach the communication channel, cannot introduce traffic into its network segment, cannot control a device connected to it, and cannot influence a component that communicates through it, then the existence of DoS as a known attack class is insufficient to establish the existence of this particular attack scenario.

2 Architecture Precedes Threats

A system architecture defines more than components and connections. It also constrains the set of interactions that are possible within the system.

Let S denote the set of system states and let

$$s_i \rightarrow s_j$$

denote a transition that is possible under the architecture, protocols, access controls, trust relationships, and physical connectivity of the system.

An attack scenario can then be regarded as a sequence of reachable transitions

$$s_0 \rightarrow s_1 \rightarrow \dots \rightarrow s_n,$$

where s_0 is a state available to the attacker and s_n is a state in which the targeted security property can be violated.

This changes the order of threat modeling.

Instead of

$$\text{catalogue} \rightarrow \text{threat} \rightarrow \text{risk assessment},$$

the analysis should proceed approximately as

$$\text{system} \rightarrow \text{architecture} \rightarrow \text{trust boundaries} \rightarrow \text{reachable states} \rightarrow \text{attack paths} \rightarrow \text{threats}.$$

Threat catalogues remain useful, but their role changes. They help verify that known mechanisms have not been overlooked; they do not define which attack scenarios exist for a particular architecture.

3 The DoS Inside the Control Network

The distinction becomes especially visible in industrial control systems.

Suppose a threat model contains the scenario

DoS attack against ICS communication channels.

The immediate temptation is to estimate its consequences: loss of telemetry, delayed commands, process interruption, degradation of control, or transition into a fail-safe state.

All of these may be important. But they describe what happens *after* the attacker has acquired the ability to interfere with the communication channel.

For a strongly isolated technological segment, this assumption is far from trivial.

An external attacker may first need to compromise another system, cross a trust boundary, obtain control over an intermediate component, reach the technological network, and only then acquire the capability to generate or suppress traffic relevant to the control process.

The actual scenario is therefore closer to

$$s_{\text{external}} \rightarrow s_{\text{compromised}} \rightarrow s_{\text{OT}} \rightarrow s_{\text{DoS}}.$$

The interesting security questions concern the arrows.

How was the first boundary crossed? Which architectural relation permitted the next transition? Which component had sufficient authority to affect the technological network? Which assumption about trust failed?

Starting the analysis directly at s_{DoS} removes precisely the part of the scenario that the threat model should explain.

4 Unreachable Is Not Low Risk

This distinction also affects quantitative risk assessment.

Suppose an integral risk model assigns a numerical value to a DoS scenario. A low value may suggest that the scenario is unlikely or relatively unimportant.

But there is a categorical difference between

a reachable attack with low probability

and

an attack requiring a state that the modeled architecture does not permit.

The latter should not merely receive a smaller coefficient.

It should first be excluded from the set of realizable attack paths, unless another scenario demonstrates how the supposedly unreachable state can be reached.

Conversely, discovering such a path is itself an important result. If an attacker can unexpectedly move from an external or less trusted environment into a protected technological segment, the security problem is not adequately described by the final attack technique. The newly discovered transition in the architecture may be more important than the DoS, MitM, command injection, or other action eventually performed through it.

5 Threat Models as Reachability Models

This suggests a simple interpretation of threat modeling.

A threat model is not primarily a list of undesirable events. It is a model of how undesirable states can become reachable.

Attack catalogues describe known mechanisms for transitions. Vulnerability databases describe conditions that may enable some of those transitions. Architecture constrains where transitions may occur. Trust boundaries constrain who is expected to perform them.

The resulting question is therefore not simply

Can this attack exist?

but

Can this system reach a state in which this attack can be performed, starting from capabilities actually available to the attacker?

This is a substantially stronger requirement.

It also prevents a common failure mode in security analysis: constructing a threat vector by selecting familiar attack names first and only afterwards attempting to fit them to the system being analyzed.

The alphabet of attacks does not define the threat model.

The architecture does.