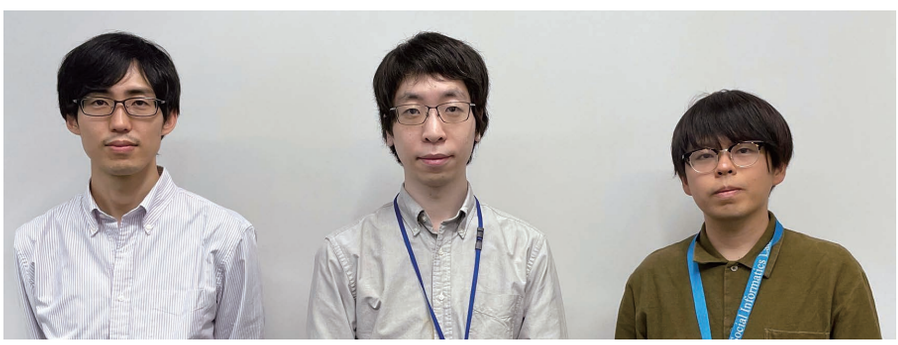
3 サイバーセキュリティ、脆弱性検知・修正

LLM時代のソフトウェア脆弱性自動修正技術

ソフトウェア開発において LLM の利用が進んでいる。開発の効率化が期待されるが、AI が生成するプログラムの安全性も担保する必要がある。このような課題の解決にむけた取り組みとして NTT 社会情報研究所が確立した脆弱性の自動修正技術と、その展開について紹介する。

LLM を活用したソフトウェア開発におけるセキュリティ上の課題



NTT 社会情報研究所
社会情報理論研究プロジェクト
サイバーセキュリティ基礎技術グループ
（左から）准特別研究員　**千田 忠賢** 氏、研究員　**上川 先之** 氏、研究員　**川田 修太郎** 氏

昨今のソフトウェア開発では、コーディング補助として LLM の利用が進んでいる。これにより開発の効率化が期待される一方で、AI が生成するプログラムに脆弱性が含まれ得るというリスクも考慮する必要がある。脆弱性とはプログラム中に存在するセキュリティ上の欠陥であり、サービス提供時に残存している脆弱性は、情報漏洩やサービス停止等の重大なインシデントの原因となる。このことから、LLM を活用した開発においても脆弱性を漏れなく検知・修正する必要がある。

現状、ソフトウェア中の脆弱性を漏れなく検知する為には網羅的な脆弱性診断が必要であり、検知した脆弱性の修正には専門的なスキルが必要である。LLM の効率性を損なうことなく安全性を担保するために、

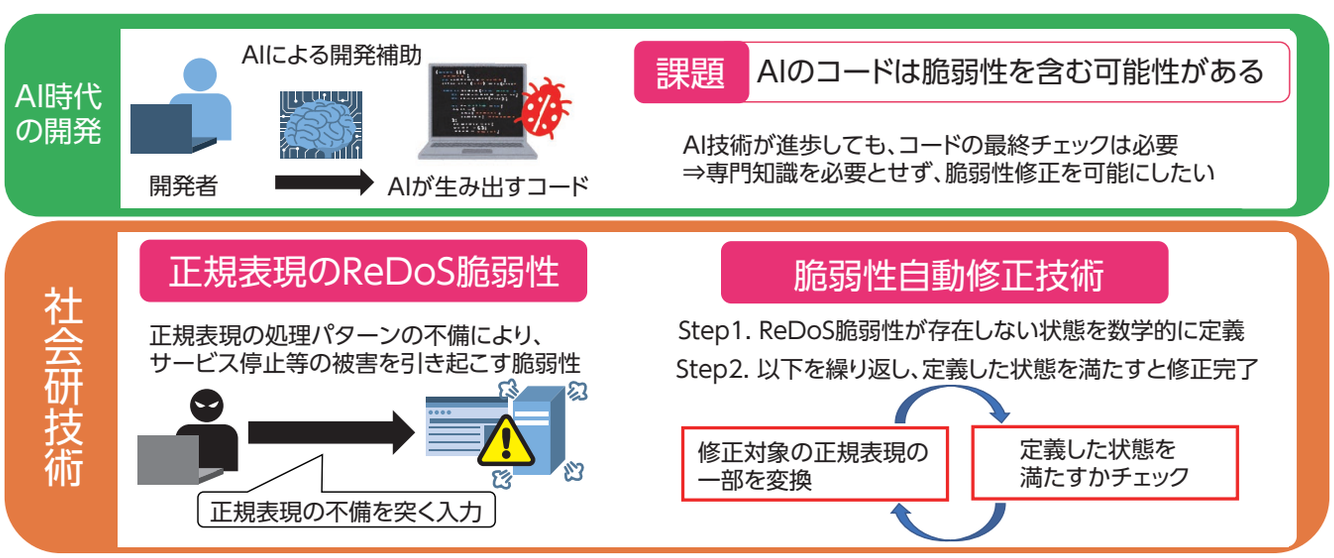


図 1　AI 時代の開発における課題と、それを解決する社会研技術

この検知・修正工程を如何に自動化するかが課題となる。

専門知識を必要としない脆弱性の自動修正

NTT 社会情報研究所では、セキュアなソフトウェア開発の実現に向けた研究を多角的に進めており、脆弱性の検知・修正工程の自動化についても取り組んできた。この取り組みの一つに、形式検証と呼ばれる数学的な手法を用いて脆弱性を修正するアプローチがあり、修正後のプログラム中に脆弱性が存在しないことを理論的に保証することをめざしている。このアプローチを適用することで、専門知識がなくても AI が生成したプログラムに含まれる脆弱性の自動修正が可能になる。

ReDoS 脆弱性（正規表現中の脆弱性）とその修正技術

この取り組みの対象として NTT 社会情報研究所は正規表現に着目した。正規表現とは文字列のパターンを表現する記法であり、さまざまなサービスで活用されている。例えばウェブ上の申し込みページでは、フォームに情報を入力した際、入力した情報の形式に誤りがあるなどのエラーが返されることがある。これはシステム側で動作している正規表現によって入力形式のチェックが行われた結果である。

このように我々の身近なところで利用されている正規表現には ReDoS と呼ばれる脆弱性が存在する。この脆弱性は正規表現の記述の不備に起因し、この脆弱性を突かれると正規表現の内部処理が正常に終了しなくなり、サーバのリソース枯渇・システム停止などの被害が発生する恐れがある。ReDoS 脆弱性の検知・修正は、正規表現エンジンの内部処理に関する深い見識を要するため、非専門家では対応が難しい。NTT 社会情報研究所は「正規表現エンジンの内部処理」「ReDoS 脆弱性が存在しない状態」を定義したうえで形式検証を用いたアプローチにより、ReDoS 脆弱性を自動で修正する技術を確立した（図 1）。これらの研究は学術的にも高く評価され、サイバーセキュリティ分野およびプログラミング言語分野の最難関国際会議として知られる IEEE Symposium on Security and Privacy（S&P'22）や ACM SIGPLAN Conference on Programming Language Design and Implementation（PLDI'23）にも採択された。

本技術は、ソフトウェア開発の検知・修正工程の担当者が入力する脆弱性の有無を確認したい正規表現および処理条件（正例・負例）に対し、ReDoS 脆弱性のない正規表現を出力することが可能である（図 2）。

今後の展望

NTT 社会情報研究所では、脆弱性自動修正技術について、ソフトウェア開発現場でのトライアルを通して、利便性向上や更なる課題抽出・改善を進めていく。また、脆弱性の根絶に向け、ReDoS 脆弱性以外の脆弱性に対象を拡大すべく、研究を進めている。LLM の台頭により大きな節目を迎えているソフトウェア開発だが、利便性が向上する一方で今後もさまざまなセキュリティリスクが表面化すると予想される。

NTT 社会情報研究所が蓄積してきたサイバーセキュリティに関する知見と理論的なアプローチを用いてこれらの課題に取り組み、LLM を活用したセキュアなソフトウェア開発に貢献していく。

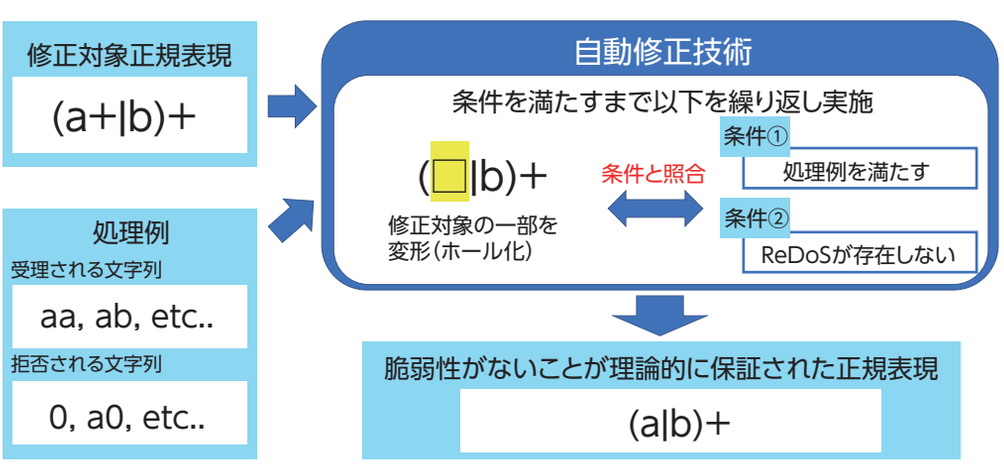


図 2　自動修正技術による正規表現の修正イメージ